Model checking interval temporal logics with regular expressions[☆]

Laura Bozzelli[a], Alberto Molinari[b], Angelo Montanari[b,*], Adriano Peron[a]

[a] *Department of Electronic Engineering and Information Technologies, University of Napoli "Federico II", Italy*
[b] *Department of Mathematics, Computer Science, and Physics, University of Udine, Italy*

Keywords:
Interval temporal logic
Model checking
Computational complexity

ABSTRACT

In this paper, we investigate the model checking (MC) problem for Halpern and Shoham's modal logic of time intervals (HS) and its fragments, where labeling of intervals is defined by regular expressions. The MC problem for HS has recently emerged as a viable alternative to the traditional (point-based) temporal logic MC. Most expressiveness and complexity results have been obtained by imposing suitable restrictions on interval labeling, namely, by either defining it in terms of interval endpoints, or by constraining a proposition letter to hold over an interval if and only if it holds over each component state (homogeneity assumption). In both cases, the expressiveness of HS gets noticeably limited, in particular when fragments of HS are considered.

A possible way to increase the expressiveness of interval temporal logic MC was proposed by Lomuscio and Michaliszyn, who suggested to use regular expressions to define interval labeling, i.e., the properties that hold true over intervals/computation stretches, based on their component points/system states. In this paper, we provide a systematic account of decidability and complexity issues for model checking HS and its fragments extended with regular expressions. We first prove that MC for (full) HS extended with regular expressions is decidable by an automaton-theoretic argument. Though the exact complexity of full HS MC remains an open issue, the complexity of all relevant proper fragments of HS is here determined. In particular, we provide an asymptotically optimal bound to the complexity of the two syntactically maximal fragments $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}E\overline{B}\overline{E}}$, by showing that their MC problem is $\mathbf{AEXP}_{\mathbf{pol}}$-complete ($\mathbf{AEXP}_{\mathbf{pol}}$ is the complexity class of problems decided by exponential-time bounded alternating Turing Machines making a polynomially bounded number of alternations). Moreover, we show that a better result holds for $\mathsf{A\overline{A}B\overline{B}}$, $\mathsf{A\overline{A}E\overline{E}}$ and all their sub-fragments, whose MC problem turns out to be **PSPACE**-complete.

1. Introduction

Modal and temporal logics are commonly used to express the properties of models of systems which are to be checked for some expected features, typically, fairness, non-starvation, state reachability, deadlock freedom, and so on. There exist algorithms which are able to automatically check temporal logic formulas over models—to ensure that the systems

[☆] This paper is an extended and revised version of [3] and [4].
[*] Corresponding author.
E-mail addresses: lr.bozzelli@gmail.com (L. Bozzelli), molinari.alberto@gmail.com (A. Molinari), angelo.montanari@uniud.it (A. Montanari),

Table 1
Complexity of MC for HS and its fragments (†local MC).

	Homogeneity [29]	Regular expressions	Endpoints [23–25]
Full HS, BE	non-elementary **EXPSPACE**-hard	non-elementary **EXPSPACE**-hard	$\mathsf{BE} + KC^\dagger$: **PSPACE** $\mathsf{BE}^\dagger$: **P**
$\mathsf{A\overline{A}B\overline{B}\overline{E}}$, $\mathsf{A\overline{A}E\overline{B}\overline{E}}$	$\in$ **EXPSPACE** $\in$ $\mathbf{AEXP_{pol}}$ **PSPACE**-hard	non-elem. **PSPACE**-hard $\mathbf{AEXP_{pol}}$-complete	
$\mathsf{A\overline{A}\overline{B}\overline{E}}$	**PSPACE**-complete	$\in$ $\mathbf{AEXP_{pol}}$ **PSPACE**-hard	
$\mathsf{A\overline{A}B\overline{B}}$, $\mathsf{B\overline{B}}$, $\mathsf{\overline{B}}$, $\mathsf{A\overline{A}E\overline{E}}$, $\mathsf{E\overline{E}}$, $\mathsf{\overline{E}}$	**PSPACE**-complete	**PSPACE**-complete	$\mathsf{A\overline{B}} + KC$: non-elem.
$\mathsf{A\overline{A}B}$, $\mathsf{A\overline{A}E}$, AB, $\mathsf{\overline{A}E}$	$\mathbf{P^{NP}}$-complete	**PSPACE**-complete	
$\mathsf{A\overline{A}}$, $\mathsf{\overline{A}B}$, AE, A, $\mathsf{\overline{A}}$	$\in \mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$ $\mathbf{P}^{\mathbf{NP}[O(\log n)]}$-hard	**PSPACE**-complete	
Prop, B, E	**co**-**NP**-complete	**PSPACE**-complete	

meet the expected behavior—searching for possible computations that violate them, i.e., determining the presence of bugs. This approach is commonly referred to as model checking (MC), which is recognized as one of the most effective techniques for automatic system verification [2]. MC has been employed also in the context of databases (e.g., active databases, database-backed web applications, and NoSQL databases) and artificial intelligence (e.g., planning, configuration systems, and multi-agent systems) [19,18,26].

A good balancing of expressiveness and complexity in the choice of the system model and the specification formalism is a key factor for the effective exploitation of MC. Systems are usually modeled as finite-state transition graphs (Kripke structures), while properties are commonly expressed by formulas of the point-based temporal logics LTL, CTL, and CTL* [35, 14]. These logics allow one to predicate properties of system *states*, and are traditionally adopted in MC as they are easy to understand and to be used also by non-experts, and suitable for practical purposes in many application domains.

Various improvements to the computational model and/or the specification language have been proposed in the literature. As for the former, we mention MC for pushdown systems (see, e.g., [15]), that feature an infinite state space, while for the latter we recall the extensions of LTL with promptness, that make it possible to bound the delay with which a liveness request is fulfilled (see, e.g., [21]). Adding regular expressions is another possible direction, that allows one to enrich the expressive power of existing logics. It has been investigated, for instance, in the cases of LTL [22] and CTL [28].

In this paper, we study the MC problem for *interval temporal logic* (ITL) extended with regular expressions. ITLs have intervals, and not points, as their primitive temporal entities [20,34,38], thus providing a different means for reasoning about time. As a matter of fact, ITLs allow one to deal with relevant temporal properties, such as actions with duration, accomplishments, and temporal aggregations, which are inherently “interval-based” and cannot be properly expressed by point-based temporal logics. ITLs have been fruitfully applied in various areas of computer science, including formal verification, computational linguistics, planning, and multi-agent systems [23,24,34]. In the last years, ITL MC has been proposed as an alternative to the traditional (point-based) temporal logic MC—which can be recovered as a special case [7,33].

Among ITLs, the most well-known one is *Halpern and Shoham's modal logic of time intervals* HS [20]. It features one modality for each of the 13 possible binary ordering relations between pairs of intervals (the so-called Allen's relations [1]), apart from equality (in fact, the three Allen's modalities A (for *meets*), B (for *started-by*), and E (for *finished-by*), together with the three modalities $\mathsf{\overline{A}}$, $\mathsf{\overline{B}}$, and $\mathsf{\overline{E}}$ for the inverse relations, suffice for expressing the entire set of relations). Its satisfiability problem is undecidable over all relevant classes of linear orders [20], and most of its fragments are undecidable as well [10,27]. Some meaningful exceptions are the logic of temporal neighborhood $\mathsf{A\overline{A}}$ and the logic of sub-intervals D [12,11].

The MC problem for HS and its fragments consists in the verification of the correctness of the behavior of a given system with respect to some relevant interval properties. To make it effective, we need to collect information about states into computation stretches: we interpret each finite computation path as an interval, and we define its labeling on the basis of the labeling of the states that compose it. Most results have been obtained by imposing suitable restrictions on interval labeling: either by constraining a proposition letter to hold over an interval if and only if it holds over each component state (homogeneity assumption [36]), or by defining interval labeling in terms of interval endpoints.

In [29], Molinari et al. deal with MC for full HS over finite Kripke structures, under the homogeneity assumption, according to a *state-based semantics* that allows branching in the past and in the future. They introduce the fundamental elements of the problem and prove its non-elementary decidability and **PSPACE**-hardness. Since then, the attention was also brought to the fragments of HS, which, similarly to what happens with satisfiability, are often computationally much better [30,31,6, 8,29,32]. The summary of these results is depicted in the second column of Table 1 (the first column reports the fragments of HS denoted by the list of the featured modalities). The complexity classes shown in red represent new (upper/lower) bounds to the complexity of the problem deriving from the results of this paper, while the other classes (in black) are known bounds (For interpretation of the references to color please refer to the web version of this article.). Only few, hard issues are left open in this picture, mostly regarding the precise complexity of the full logic and its maximal fragments. A comparison of alternative semantics, that is, state-based, trace-based, and computation-tree-based semantics, together with an expressiveness comparison with standard point-based temporal logics LTL, CTL, and CTL* can be found in [7].

Different assumptions have been done by Lomuscio and Michaliszyn in [23,24] for some HS fragments extended with epistemic operators (KC). They assume a *computation-tree-based semantics* (formulae are interpreted over the unwinding of the Kripke structure) and interval labeling takes into account only the endpoints of intervals. The different semantic assumptions prevent a direct comparison with the former approach. The decidability status of MC for full epistemic HS is still unknown. A summary of the results by Lomuscio and Michaliszyn is depicted in the last column of Table 1.

The first meaningful attempt to relax the homogeneity assumption can be found in [25], where Lomuscio and Michaliszyn propose to use regular expressions to define the labeling of proposition letters over intervals in terms of the component states (notice that homogeneity can be trivially encoded by regular expressions). In that work the authors prove the decidability of MC with regular expressions for some very restricted fragments of epistemic HS, giving some rough upper bounds to its computational complexity.

In this paper, we give a detailed picture of decidability and complexity for HS with regular expressions, which was still missing. The results are summarized in the third column of Table 1. It is interesting to compare the complexity of MC for HS fragments extended with regular expressions with the same fragments under the homogeneity assumption. The rich spectrum of complexities for the less expressive fragments of HS under homogeneity (last four rows in the table) collapses to **PSPACE**-completeness in the case of the corresponding fragments with regular expressions, witnessing that using regular expressions increases the expressive power of (syntactically) small fragments of HS. Whether or not there exists an elementary algorithm for full HS remains an open issue, just like in the case of full HS under homogeneity. The main results of the paper are summarized in the following short account of the contents of the next sections.

In Section 2, we introduce the logic HS, along with some background knowledge. Then, in Section 3, we summarize known complexity results about MC for HS and its fragments. Next, in Section 4, we prove that MC for full HS extended with regular expressions, under the state-based semantics, is decidable, by exploiting an automaton-theoretic technique. Then, in Section 5, we study the problem of MC for the two syntactically maximal (symmetric) fragments $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}E\overline{B}\overline{E}}$ with regular expressions, proving that it is $\mathbf{AEXP_{pol}}$-complete ($\mathbf{AEXP_{pol}}$ denotes the complexity class of problems decided by exponential-time bounded alternating Turing Machines with a polynomially bounded number of alternations). Such a class captures the exact complexity of some relevant problems [9,16], like, for instance, the first-order theory of real addition with order [16]. Finally, in Section 6, we show that formulas of a large class of HS fragments, i.e., those featuring (any subset of) HS modalities for the Allen's relations *meets, met-by, started-by*, and *starts* ($\mathsf{A\overline{A}B\overline{B}}$), can be checked in polynomial working space (MC for all these is **PSPACE**-complete). Conclusions provide an assessment of the work done, and outline future research directions.

2. Preliminaries

In this section, we introduce the interval logic HS together with some notation and background knowledge about Kripke structures, regular expressions, and finite state automata.

Let $\mathbb{N}$ be the set of natural numbers. For all $i, j \in \mathbb{N}$, we denote by $[i, j]$, with $i \leq j$, the set of naturals h such that $i \leq h \leq j$.

Let Σ be an alphabet, w be a non-empty finite word over Σ, and ε be the empty word. We denote by $|w|$ the length of w. For all $1 \leq i \leq j \leq |w|$, $w(i)$ represents the i-th letter of w (i is called a w-position), while $w(i, j)$ is the finite subword of w given by $w(i) \cdots w(j)$. Let $|w| = n$. We define $\mathrm{fst}(w) = w(1)$ and $\mathrm{lst}(w) = w(n)$. The sets of all proper prefixes (resp., suffixes) of w is $\mathrm{Pref}(w) = \{w(1, i) \mid 1 \leq i \leq n-1\}$ (resp., $\mathrm{Suff}(w) = \{w(i, n) \mid 2 \leq i \leq n\}$). For $i \in [1, n]$, w^i is a shorthand for $w(1, i)$. The concatenation of two words w and w' is denoted as usual by $w \cdot w'$. Moreover, if $\mathrm{lst}(w) = \mathrm{fst}(w')$, $w \star w'$ stands for $w(1, n-1) \cdot w'$.

For all $h, n \geq 0$, let $\mathsf{Tower}(h, n)$ denote a tower of exponentials of height h and argument n, that is, $\mathsf{Tower}(0, n) = n$ and $\mathsf{Tower}(h+1, n) = 2^{\mathsf{Tower}(h,n)}$. Moreover, let h-**EXPTIME** denote the class of languages decided by deterministic Turing machines whose number of computation steps is bounded by functions of n in $O(\mathsf{Tower}(h, n^c))$, for some constant $c \geq 1$. Note that 0-**EXPTIME** is **P**.

2.1. Kripke structures, regular expressions, and finite automata

Finite state systems are usually modeled as finite Kripke structures. Let $\mathcal{AP}$ be a finite set of proposition letters, which represent predicates over the states of the considered system.

Definition 1 *(Kripke structure).* A *Kripke structure* is a tuple $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$, where S is a set of states, $R \subseteq S \times S$ is a left-total transition relation, $\mu : S \mapsto 2^{\mathcal{AP}}$ is a total labeling function assigning to each state s the set of proposition letters

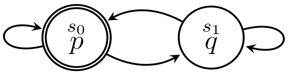


Fig. 1. The Kripke structure $\mathcal{K}_2$.

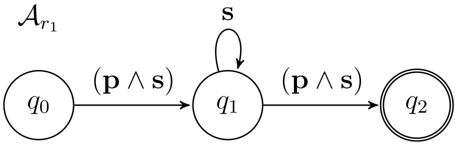

$\mathcal{A}_{r_1}$
$\mathbf{s}$
$(\mathbf{p} \wedge \mathbf{s})$
$(\mathbf{p} \wedge \mathbf{s})$
q_0
q_1
q_2

Fig. 2. The canonical NFA $\mathcal{A}_{r_1}$ associated with the regular expression r_1 of Example 2.

that hold over it, and $s_0 \in S$ is the initial state. For $s \in S$, the set $R(s)$ of successors of s is the non-empty set of states s' such that $(s, s') \in R$. We say that $\mathcal{K}$ is finite if S is finite.

Fig. 1 depicts a finite Kripke structure $\mathcal{K}_2 = (\{p, q\}, \{s_0, s_1\}, R, \mu, s_0)$, where $R = \{(s_i, s_j) \mid i, j = 0, 1\}$, $\mu(s_0) = \{p\}$, $\mu(s_1) = \{q\}$. The initial state s_0 is marked by a double circle.

Let $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ be a Kripke structure. A *trace* of $\mathcal{K}$ is a non-empty finite word ρ over S such that $(\rho(i), \rho(i+1)) \in R$ for $i \in [1, |\rho| - 1]$. A trace is *initial* if it starts from s_0. We denote by $\mathrm{Trc}_{\mathcal{K}}$ the *infinite* set of traces of $\mathcal{K}$. A trace ρ induces a *labeling sequence*, namely, the finite word $\mu(\rho)$ over $2^{\mathcal{AP}}$ given by $\mu(\rho(1)) \ldots \mu(\rho(n))$, with $n = |\rho|$.

Let us now recall the notion of regular expressions over finite words. Since we are interested in expressing requirements over the labeling sequences induced by traces — i.e., finite words over $2^{\mathcal{AP}}$—here we consider *proposition-based* regular expressions (denoted as REs), where atomic expressions are propositional formulas over $\mathcal{AP}$—instead of letters over an alphabet. Formally, the set of REs r over $\mathcal{AP}$ is defined by the grammar:

$$r ::= \varepsilon \mid \phi \mid r \cup r \mid r \cdot r \mid r^*,$$

where ϕ is a propositional formula over $\mathcal{AP}$. The length $|r|$ of an RE r is the number of subexpressions of r. An RE r denotes a language $\mathcal{L}(r)$ of finite words over $2^{\mathcal{AP}}$ defined as:

- $\mathcal{L}(\varepsilon) = \{\varepsilon\}$,
- $\mathcal{L}(\phi) = \{A \in 2^{\mathcal{AP}} \mid A \text{ satisfies } \phi\}$,
- $\mathcal{L}(r_1 \cup r_2) = \mathcal{L}(r_1) \cup \mathcal{L}(r_2)$,
- $\mathcal{L}(r_1 \cdot r_2) = \mathcal{L}(r_1) \cdot \mathcal{L}(r_2)$, and
- $\mathcal{L}(r^*) = (\mathcal{L}(r))^*$.

By well-known results, the class of RE over $\mathcal{AP}$ captures the class of regular languages of finite words over $2^{\mathcal{AP}}$.

Example 2. An example of RE is $r_1 = (\mathbf{p} \wedge \mathbf{s}) \cdot \mathbf{s}^* \cdot (\mathbf{p} \wedge \mathbf{s})$ that intuitively denotes the set of finite words where both $\mathbf{p}$ and $\mathbf{s}$ hold true on the endpoints, and $\mathbf{s}$ continuously holds in all internal symbols/sets of $2^{\mathcal{AP}}$. The RE $r_2 = (\neg \mathbf{p})^*$ denotes the set of finite words such that $\mathbf{p}$ does not hold in any position.

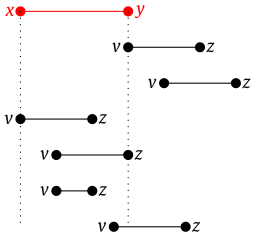
We also recall the standard notion of *non-deterministic finite state automaton* (NFA), which is a tuple $\mathcal{A} = (\Sigma, Q, Q_0, \Delta, F)$, where Σ is a finite alphabet, Q is a finite set of states, $Q_0 \subseteq Q$ is the set of initial states, $\Delta : Q \times \Sigma \mapsto 2^Q$ is the transition function (or, equivalently, $\Delta \subseteq Q \times \Sigma \times Q$), and $F \subseteq Q$ is the set of accepting states. An NFA $\mathcal{A}$ is *complete* if, for all $(q, \sigma) \in Q \times \Sigma$, $\Delta(q, \sigma) \neq \emptyset$. Given a finite word w over Σ, with $|w| = n$, and two states $q, q' \in Q$, a *run* (or *computation*) of $\mathcal{A}$ from q to q' over w is a finite sequence of states $q_1, \ldots, q_{n+1}$ such that $q_1 = q$, $q_{n+1} = q'$, and $q_{i+1} \in \Delta(q_i, w(i))$ for all $i \in [1, n]$. The language $\mathcal{L}(\mathcal{A})$ *accepted by* $\mathcal{A}$ consists of the finite words w over Σ such that there is a run over w from some initial state to some accepting state.

A deterministic finite state automaton (DFA) is an NFA $\mathcal{D} = (\Sigma, Q, Q_0, \Delta, F)$ such that Q_0 is a singleton, and for all $(q, c) \in Q \times \Sigma$, $\Delta(q, c)$ is a singleton. In the following, in the case of a DFA, we will denote the transition function Δ as δ.

Remark 3. By well-known results, given an RE r over $\mathcal{AP}$, one can construct, in a compositional way, an NFA $\mathcal{A}_r$ with alphabet $2^{\mathcal{AP}}$, whose number of states is at most $2|r|$, such that $\mathcal{L}(\mathcal{A}_r) = \mathcal{L}(r)$. We call $\mathcal{A}_r$ the *canonical* NFA associated with r.

Note that, though the number of edges of $\mathcal{A}_r$ may be exponential in $|\mathcal{AP}|$ (edges are labeled by assignments $A \in 2^{\mathcal{AP}}$ satisfying propositional formulas ϕ of r), we can avoid to explicitly store edges, as they can be recovered in polynomial time from r. In Fig. 2, we depict the canonical NFA $\mathcal{A}_{r_1}$ associated with the RE r_1 of Example 2. We can avoid storing the edges of $\mathcal{A}_{r_1}$ by remembering which propositional formulas of r_1 they are associated with.

Table 2
Allen's relations and corresponding HS modalities.

Allen relation	HS	Definition w.r.t. interval structures	Example
MEETS	$\langle A \rangle$	$[x, y]\mathcal{R}_A[v, z] \iff y = v$	
BEFORE	$\langle L \rangle$	$[x, y]\mathcal{R}_L[v, z] \iff y < v$	
STARTED-BY	$\langle B \rangle$	$[x, y]\mathcal{R}_B[v, z] \iff x = v \wedge z < y$	
FINISHED-BY	$\langle E \rangle$	$[x, y]\mathcal{R}_E[v, z] \iff y = z \wedge x < v$	
CONTAINS	$\langle D \rangle$	$[x, y]\mathcal{R}_D[v, z] \iff x < v \wedge z < y$	
OVERLAPS	$\langle O \rangle$	$[x, y]\mathcal{R}_O[v, z] \iff x < v < y < z$	

2.2. The interval temporal logic HS

An interval algebra to reason about intervals and their relative order was proposed by Allen in [1], while a systematic logical study of interval representation and reasoning was done a few years later by Halpern and Shoham, who introduced the interval temporal logic HS featuring one modality for each Allen relation, but equality [20]. Table 2 depicts 6 of the 13 Allen's relations, together with the corresponding HS (existential) modalities. The other 7 relations are the 6 inverse relations (the inverse $\overline{\mathcal{R}}$ of a binary relation $\mathcal{R}$ is such that $b\overline{\mathcal{R}}a$ if and only if $a\mathcal{R}b$) and equality. If $\langle X \rangle$ is the modality for $\mathcal{R}$, $\langle \overline{X} \rangle$ is the modality for $\overline{\mathcal{R}}$.

Let $\mathcal{P}_u$ be a finite set of *uninterpreted interval properties*. The HS language over $\mathcal{P}_u$ consists of proposition letters from $\mathcal{P}_u$, the Boolean connectives $\neg$ and $\wedge$, and a temporal modality for each of the (non trivial) Allen's relations, i.e., $\langle A \rangle$, $\langle L \rangle$, $\langle B \rangle$, $\langle E \rangle$, $\langle D \rangle$, $\langle O \rangle$, $\langle \overline{A} \rangle$, $\langle \overline{L} \rangle$, $\langle \overline{B} \rangle$, $\langle \overline{E} \rangle$, $\langle \overline{D} \rangle$, and $\langle \overline{O} \rangle$. HS formulas are defined by the grammar

$$\psi ::= p_u \mid \neg\psi \mid \psi \wedge \psi \mid \langle X \rangle \psi \mid \langle \overline{X} \rangle \psi,$$

where $p_u \in \mathcal{P}_u$ and $X \in \{A, L, B, E, D, O\}$. We will also use the other standard connectives (disjunction $\vee$ and implication $\rightarrow$). Moreover, for any modality X, the dual universal modalities $[X]\psi$ and $[\overline{X}]\psi$ are defined as $\neg\langle X \rangle\neg\psi$ and $\neg\langle \overline{X} \rangle\neg\psi$, respectively.

Given any subset of Allen's relations $\{X_1, \ldots, X_n\}$, we denote by $\mathsf{X_1 \cdots X_n}$ the HS fragment that features existential (and universal) modalities for $X_1, \ldots, X_n$ only.

W.l.o.g., we assume the *non-strict semantics of HS*, which admits intervals consisting of a single point.[1] Under such an assumption, all HS modalities can be expressed in terms of modalities $\langle B \rangle$, $\langle E \rangle$, $\langle \overline{B} \rangle$, and $\langle \overline{E} \rangle$ [38]. As an example, modality $\langle A \rangle$ can be expressed in terms of $\langle E \rangle$ and $\langle \overline{B} \rangle$ as follows: $\langle A \rangle \varphi := ([E]\bot \wedge (\varphi \vee \langle \overline{B} \rangle \varphi)) \vee \langle E \rangle([E]\bot \wedge (\varphi \vee \langle \overline{B} \rangle \varphi))$. We observe that $[E]\bot$ is true only on point-intervals, requiring that no suffix of the current interval exists. HS can thus be viewed as a multi-modal logic with 4 primitive modalities. However, since in the following we will focus on some HS fragments which do not feature (some of) $\langle B \rangle$, $\langle \overline{B} \rangle$, $\langle E \rangle$, and $\langle \overline{E} \rangle$, we explicitly add both $\langle A \rangle$ and $\langle \overline{A} \rangle$ to the considered set of modalities.

In [29], the authors investigate the MC problem over finite Kripke structures $\mathcal{K}$ for HS formulas where intervals correspond to the traces of $\mathcal{K}$. The approach followed there is subject to two restrictions: (*i*) the set $\mathcal{P}_u$ of HS proposition letters and the set $\mathcal{AP}$ of proposition letters for the Kripke structure coincide, and (*ii*) a proposition letter holds over an interval if and only if it holds over all its sub-intervals (*homogeneity assumption*). Here, we adopt a more general and expressive approach according to which an abstract interval proposition letter $p_u \in \mathcal{P}_u$ denotes a regular language of finite words over $2^{\mathcal{AP}}$. More specifically, every p_u is a (proposition-based) regular expression over $\mathcal{AP}$. Thus, hereafter, an HS formula φ over $\mathcal{AP}$ is an HS formula whose interval proposition letters (or atomic formulas) are REs r over $\mathcal{AP}$. For this reason, we define the size (or length) $|\varphi|$ of φ as the number of non-atomic subformulas of φ plus $\sum_{r \in \mathsf{spec}} |r|$, where spec is the set of REs occurring in φ.

Given a Kripke structure $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$, a trace ρ of $\mathcal{K}$, and an HS formula φ over $\mathcal{AP}$, the satisfaction relation $\mathcal{K}, \rho \models \varphi$ is inductively defined as follows (we omit the standard clauses for Boolean connectives):

- $\mathcal{K}, \rho \models r$ if and only if $\mu(\rho) \in \mathcal{L}(r)$ for each RE r over $\mathcal{AP}$,
- $\mathcal{K}, \rho \models \langle B \rangle \varphi$ if and only if there exists $\rho' \in \text{Pref}(\rho)$ such that $\mathcal{K}, \rho' \models \varphi$,
- $\mathcal{K}, \rho \models \langle E \rangle \varphi$ if and only if there exists $\rho' \in \text{Suff}(\rho)$ such that $\mathcal{K}, \rho' \models \varphi$,
- $\mathcal{K}, \rho \models \langle \overline{B} \rangle \varphi$ if and only if $\mathcal{K}, \rho' \models \varphi$ for some trace ρ' such that $\rho \in \text{Pref}(\rho')$,
- $\mathcal{K}, \rho \models \langle \overline{E} \rangle \varphi$ if and only if $\mathcal{K}, \rho' \models \varphi$ for some trace ρ' such that $\rho \in \text{Suff}(\rho')$.

[1] All the results we prove in the paper hold for the strict semantics as well.

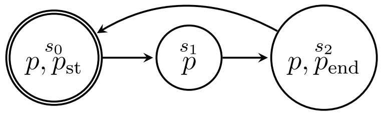


Fig. 3. Kripke structure representing a printer.

$\mathcal{K}$ is a *model* of φ, denoted as $\mathcal{K} \models \varphi$, if for all *initial* traces ρ of $\mathcal{K}$, it holds that $\mathcal{K}, \rho \models \varphi$. The *MC problem* for HS is the problem of checking, for a finite Kripke structure $\mathcal{K}$ and an HS formula φ, whether or not $\mathcal{K} \models \varphi$. The problem is not trivially decidable since the set $\mathrm{Trc}_{\mathcal{K}}$ of traces of $\mathcal{K}$ is infinite.

Note that the considered state-based semantics provides a branching-time setting both in the past and in the future. In particular, while the modalities for B and E are linear-time (as they allow us to select prefixes and suffixes of the current trace only), the modalities for A and $\overline{B}$ (respectively, $\overline{A}$ and $\overline{E}$) are branching-time in the future (respectively, in the past) since they enable us to nondeterministically extend a trace in the future (respectively, in the past). As shown in [7], for the considered semantics, the logics HS and CTL^* are expressively incomparable already under the homogeneity assumption. However, under the homogeneity assumption, the use of the past branching-time modalities $\overline{A}$ and $\overline{E}$ is necessary for capturing requirements which cannot be expressed in CTL^*. For instance, the constraint "*each state reachable from the initial one where p holds has a predecessor where p holds as well*" cannot be expressed in CTL^*, but can be easily stated in the fragment $\mathsf{\overline{A}E}$ [7]. Conversely, in the more expressive setting based on regular expressions, the future branching-time modalities A and $\overline{B}$ are already sufficient for capturing requirements which cannot be expressed in CTL^*, such as the following branching-time bounded response property: "*for each state, reachable from the initial one, where a request* req *occurs, there is a computation, starting from this state, such that the request is followed by a response* res *after an* even number *of steps*". This requirement can be expressed in the fragment $\mathsf{A\overline{B}}$ as follows: $[A](req \to \langle \overline{\mathrm{B}} \rangle (req \cdot (\top \cdot \top)^* \cdot res))$. Notice that it says nothing about the possible occurrence of a response *res* after an odd number of steps.

Example 4 *(Adapted from [25]).* With this (toy) example we want to compare the effectiveness of regular expressions as rules for defining interval labeling, to homogeneity and to endpoint-based labeling.

In Fig. 3, a Kripke structure representing a printer is shown. In s_0 the printer starts printing a sheet; in s_1 the process is ongoing, and it ends in s_2. The printer then prints the subsequent sheet, by "moving" back to s_0.

Imagine we want to label the process of printing a *single sheet* by p (i.e., only the trace $s_0 s_1 s_2$). Under homogeneity, if $s_0 s_1 s_2$ is labeled by p, then s_0, s_1, s_2, $s_0 s_1$, $s_1 s_2$ must be all labeled by p as well, against our idea. In the endpoint-based approach [23,24], in order for p to label $s_0 s_1 s_2$, p must also label all traces $(s_0 s_1 s_2)^n$ for $n \in \mathbb{N}^+$, as the endpoints of all these are s_0, s_2. Thus "several, consecutive sheets" are labeled p.

Conversely, we can just write the proposition-based RE $\mathbf{p}_{\text{st}} \cdot (\neg \mathbf{p}_{\text{end}} \wedge \neg \mathbf{p}_{\text{st}})^* \cdot \mathbf{p}_{\text{end}}$ to capture precisely the trace $s_0 s_1 s_2$.

3. The general picture

In this section, we give a short account of research on MC for HS and its fragments, and we enlighten the original contributions of the present paper (we refer the reader again to Table 1).

Let us consider first the MC problem for HS and its fragments, under the homogeneity assumption, according to a state-based semantics [7]. We preliminarily notice that in our setting it is easy to force homogeneity by simply imposing that all regular expressions in the formula have the form $p \cdot p^*$, for some $p \in \mathcal{AP}$.

In [29], Molinari et al. provide a MC algorithm for (full) HS, with non-elementary complexity, that, given a finite Kripke structure $\mathcal{K}$ and a bound k on the nesting depth of $\langle \mathrm{E} \rangle$ and $\langle \mathrm{B} \rangle$ modalities in the input HS formula, exploits a *finite* and satisfiability-equivalent representation for the infinite set $\mathrm{Trc}_{\mathcal{K}}$, that accounts for $\mathcal{K}$ and k. **EXPSPACE**-hardness of MC for BE, and thus for full HS, has been shown in [6]. An **EXPSPACE** MC algorithm for the fragments $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}E\overline{B}\overline{E}}$ has been devised in [31]: for any trace of $\mathcal{K}$, it finds a satisfiability-preserving trace of bounded length (*trace representative*). In this way, the MC algorithm only needs to check traces with a bounded maximum length. **PSPACE**-hardness of MC for $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}E\overline{B}\overline{E}}$ has been proved in [30]. A number of well-behaved HS fragments, which are still expressive enough to capture meaningful interval properties of state transition systems, and whose MC problem has a computational complexity markedly lower than that of full HS, have been identified in [6,8,30,32]. In particular, MC has been proved to be (i) **PSPACE**-complete for $\mathsf{A\overline{A}\overline{B}\overline{E}}$, $\mathsf{A\overline{A}B\overline{B}}$, $\mathsf{A\overline{A}E\overline{E}}$, $\mathsf{\overline{B}}$, and $\mathsf{\overline{E}}$, (ii) $\mathbf{P}^{\mathbf{NP}}$-complete for AB, $\mathsf{A\overline{A}B}$, $\mathsf{\overline{A}E}$, and $\mathsf{A\overline{A}E}$, (iii) in between $\mathbf{P}^{\mathbf{NP}[O(\log n)]}$ and $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$ [37] for $\mathsf{A\overline{A}}$, A, $\mathsf{\overline{A}}$, $\mathsf{\overline{A}B}$, and AE, and (iv) **co**-**NP**-complete for B, E, and the pure propositional fragment Prop.

In [23,24], Lomuscio and Michaliszyn investigate MC for some HS fragments extended with the epistemic modalities K and C, according to a computation-tree-based semantics [7], under the assumption that interval labeling is defined by interval endpoints only. They prove that *local* MC for $\mathsf{BE} + KC$ is **PSPACE**-complete (it is in $\mathbf{P}$ for BE), and they give a non-elementary upper bound to the complexity of MC for $\mathsf{A\overline{B}} + KC$. Notice that labeling of intervals by endpoints can be easily captured in our setting by regular expressions having the form:

$$\bigcup_{(i,j)\in I} (q_i \cdot \top^* \cdot q_j) \cup \bigcup_{i\in I'} q_i,$$

for some suitable sets of indexes $I \subseteq \{1,\ldots,|S|\}^2$ and $I' \subseteq \{1,\ldots,|S|\}$, where $q_i \in \mathcal{AP}$ is a letter labeling the state $s_i \in S$ of $\mathcal{K}$, only.

Later [25], Lomuscio and Michaliszyn propose an alternative definition of interval labeling for the two fragments, which associates a regular expression over the set of states of the Kripke structure with each proposition letter, that leads to a significant increase in expressiveness—as the labeling of an interval is no more determined by its endpoints, but it depends on the ordered sequence of states the interval consists of—at no extra computational cost. No result is presented about MC for full HS (with or without K, C).

In this paper, we define interval labeling via regular expressions in a way that can be shown to be equivalent to that of [25]. We first show, in Section 4, that MC for (full) HS extended with regular expressions (under the state-based semantics) is decidable, by exploiting an automata-theoretic approach and the notion of $\mathcal{K}$-NFA, a particular version of NFA. Moreover, the problem can be shown to be in **P** when it is restricted to system models assuming the formula to be of constant length.

Then, in Section 5, we study the problems of MC for the two (syntactically) maximal (symmetric) fragments $A\overline{A}B\overline{B}\overline{E}$ and $A\overline{A}E\overline{B}\overline{E}$ with regular expressions, proving that both problems are $\mathbf{AEXP_{pol}}$-complete. First, we note that settling the exact complexity of these fragments under the homogeneity assumption—which can be encoded by regular expressions—is a difficult open question [31]. Moreover, considering that $\mathbf{AEXP_{pol}} \subseteq$ **EXPSPACE** and that HS under homogeneity is subsumed by HS with regular expressions, the results proved in this paper improve the upper bounds for the fragments $A\overline{A}B\overline{B}\overline{E}$ and $A\overline{A}E\overline{B}\overline{E}$ given in [31]. More in detail, we preliminarily establish an exponential small-model property for $A\overline{A}B\overline{B}\overline{E}$ (Section 5.1): for each interval, it is possible to find an interval of bounded exponential length that is indistinguishable with respect to the fulfillment of $A\overline{A}B\overline{B}\overline{E}$ formulas (respectively, $A\overline{A}E\overline{B}\overline{E}$ formulas). Such a property allows us to devise an MC procedure belonging to the class $\mathbf{AEXP_{pol}}$ (Section 5.2). Finally, the matching lower bounds are obtained by polynomial-time reductions from the so-called alternating multi-tiling problem, showing that they already hold for the fragments $B\overline{E}$ and $E\overline{B}$ of $A\overline{A}B\overline{B}\overline{E}$ and $A\overline{A}E\overline{B}\overline{E}$, respectively (Section 5.3).

Finally, in Section 6, we show that formulas of HS fragments featuring (any subset of) HS modalities for the Allen's relations *meets, met-by, started-by*, and *starts* ($A\overline{A}B\overline{B}$) can be checked in polynomial working space (MC for all these is **PSPACE**-complete). In particular, in Section 6.1 we prove a small-model theorem for the fragment $A\overline{A}B\overline{B}$ (and the symmetric fragment $A\overline{A}E\overline{E}$), which is then exploited in Sections 6.2 and 6.3 to devise a **PSPACE** MC algorithm for $A\overline{A}B\overline{B}$ (and $A\overline{A}E\overline{E}$). Moreover, in Section 6.3, we prove that MC for the purely propositional fragment of HS, denoted as Prop, is hard for **PSPACE**, which is enough to conclude that MC for any sub-fragment of $A\overline{A}B\overline{B}$ or $A\overline{A}E\overline{E}$ is complete for **PSPACE**. Hence, relaxing the homogeneity assumption via regular expressions comes at no cost for $A\overline{A}B\overline{B}$, $A\overline{A}E\overline{E}$, $B\overline{B}$, $E\overline{E}$, $\overline{B}$, and $\overline{E}$—that remain in **PSPACE**—while $A\overline{A}B$ and $A\overline{A}E$ and their sub-fragments increase their complexity to **PSPACE** (see Table 1 once more).

The listed results allow us to outline a correspondence with the previously mentioned complexity results in [25] for the fragment $\mathsf{BE} + KC$ under the computation-tree-based semantics. Notice that the computation-tree-based semantics and the state-based one behave exactly in the same way when HS is restricted to fragments featuring present and future modalities only.[2] From the **PSPACE**-completeness of $A\overline{A}B\overline{B}$, it immediately follows the **PSPACE** membership of $A\overline{B}$ with regular expressions, devoid of epistemic operators (in fact, the non-elementary complexity of MC for $A\overline{B}$ in [25] can be hardly ascribed to the addition of epistemic operators).

4. MC for full HS

In this section, we develop an automata-theoretic approach to the MC problem for full HS with regular expressions. Given a finite Kripke structure $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ and an HS formula φ over $\mathcal{AP}$, we compositionally construct an NFA over the set of states S of $\mathcal{K}$ accepting the set of traces ρ of $\mathcal{K}$ such that $\mathcal{K}, \rho \models \varphi$. The size of the resulting NFA is nonelementary, but it is just *linear in the size of* $\mathcal{K}$. To prove that the nonelementary blow-up does not depend on $\mathcal{K}$, we introduce a special subclass of NFAs, called $\mathcal{K}$-NFA, which intuitively represents the "synchronization" of an NFA with the Kripke structure $\mathcal{K}$. In this way, a $\mathcal{K}$-NFA may only accept traces of $\mathcal{K}$.

Definition 5 *($\mathcal{K}$-NFA).* A $\mathcal{K}$-NFA is an NFA $\mathcal{A} = (S, Q, Q_0, \delta, F)$ over S satisfying the following conditions: *(i)* the set Q of states has the form $M \times S$ (M is called the *main component* or the set of *main states*); *(ii)* $Q_0 \cap F = \emptyset$, that is, the empty word ε is not accepted; *(iii)* for all $(q, s) \in M \times S$ and $s' \in S$, we have $\delta((q, s), s') = \emptyset$ if $s' \neq s$, and $\delta((q, s), s) \subseteq M \times R(s)$.

It is worth noticing that, for all words $\rho \in S^+$, if there is a run of the $\mathcal{K}$-NFA over ρ, then ρ is a trace of $\mathcal{K}$. In the following, we construct a $\mathcal{K}$-NFA $\mathcal{A}$ accepting the traces ρ of $\mathcal{K}$ such that $\mathcal{K}, \rho \models \varphi$.

[2] As shown in [7], this is not the case in general: the computation-tree-based semantics of [23–25] is subsumed by the state-based one of [29] and follow-up papers.

In a standard automata-theoretic approach, an automaton accepting the set of models of φ would be first defined, and then intersected with $\mathcal{K}$. In the following construction, the synchronization with $\mathcal{K}$ is instead implicitly associated with the construction of the $\mathcal{K}$-NFA itself. Such a choice is motivated by the fact that proposition letters in the formula φ (the base case in the construction) are regular expressions which have to be synchronized with the traces of $\mathcal{K}$. Such a synchronization is then maintained along the whole process of $\mathcal{K}$-NFA construction.

The recursive step for dealing with negation in φ is noteworthy, since it is not just a pure complementation of the $\mathcal{K}$-NFA under construction. As a matter of fact, only the synchronized NFA-component (for the regular expressions of φ) has to be complemented, whereas the synchronized $\mathcal{K}$-component does not. For this reason, the size of the final $\mathcal{K}$-NFA is nonelementary, but it is linear in the size of $\mathcal{K}$.

In order to prove the main result of the section (stated in Theorem 9), we preliminarily describe the composition steps to build the required $\mathcal{K}$-NFA. In particular, (i) in Proposition 6 we give the basic step to deal with propositions associated with regular expressions, (ii) in Proposition 7 the closure of $\mathcal{K}$-NFAs under language operations corresponding to HS modalities, and (iii) in Proposition 8 the closure $\mathcal{K}$-NFAs under Boolean operations.

In the following, let $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ be a finite Kripke structure over $\mathcal{AP}$.

Proposition 6. *Let $\mathcal{A}$ be an NFA over $2^{\mathcal{AP}}$ with n states. One can construct, in polynomial time, a $\mathcal{K}$-NFA $\mathcal{A}_{\mathcal{K}}$ with at most $n+1$ main states accepting the set of traces ρ of $\mathcal{K}$ such that $\mu(\rho) \in \mathcal{L}(\mathcal{A})$.*

Proof. Let $\mathcal{A} = (2^{\mathcal{AP}}, Q, Q_0, \delta, F)$. By using an additional state, we can assume $\varepsilon \notin \mathcal{L}(\mathcal{A})$, that is, $Q_0 \cap F = \emptyset$. Then, $\mathcal{A}_{\mathcal{K}} = (S, Q \times S, Q_0 \times S, \delta', F \times S)$, where for all $(q,s) \in Q \times S$ and $s' \in S$, it holds that $\delta'((q,s),s') = \emptyset$ if $s' \neq s$, and $\delta'((q,s),s) = \delta(q, \mu(s)) \times R(s)$. Since $R(s) \neq \emptyset$ for all $s \in S$, the thesis follows. $\square$

We now define the operations on languages of finite words over S corresponding to the HS modalities $\langle \mathrm{B} \rangle$, $\langle \overline{\mathrm{B}} \rangle$, $\langle \mathrm{E} \rangle$, and $\langle \overline{\mathrm{E}} \rangle$. Given a language $\mathcal{L}$ over S, we define the following languages of traces of $\mathcal{K}$:

- $\langle \mathrm{B} \rangle_{\mathcal{K}}(\mathcal{L}) = \{\rho \in \mathrm{Trc}_{\mathcal{K}} \mid \exists \rho' \in \mathcal{L} \cap S^+ \text{ and } \rho'' \in S^+ \text{ such that } \rho = \rho' \cdot \rho''\}$;
- $\langle \overline{\mathrm{B}} \rangle_{\mathcal{K}}(\mathcal{L}) = \{\rho \in \mathrm{Trc}_{\mathcal{K}} \mid \exists \rho' \in S^+ \text{ such that } \rho \cdot \rho' \in \mathcal{L} \cap \mathrm{Trc}_{\mathcal{K}}\}$;
- $\langle \mathrm{E} \rangle_{\mathcal{K}}(\mathcal{L}) = \{\rho \in \mathrm{Trc}_{\mathcal{K}} \mid \exists \rho'' \in \mathcal{L} \cap S^+ \text{ and } \rho' \in S^+ \text{ such that } \rho = \rho' \cdot \rho''\}$;
- $\langle \overline{\mathrm{E}} \rangle_{\mathcal{K}}(\mathcal{L}) = \{\rho \in \mathrm{Trc}_{\mathcal{K}} \mid \exists \rho' \in S^+ \text{ such that } \rho' \cdot \rho \in \mathcal{L} \cap \mathrm{Trc}_{\mathcal{K}}\}$.

We show that $\mathcal{K}$-NFAs are closed under the above-defined language operations $\langle \mathrm{B} \rangle_{\mathcal{K}}(\cdot)$, $\langle \mathrm{E} \rangle_{\mathcal{K}}(\cdot)$, $\langle \overline{\mathrm{B}} \rangle_{\mathcal{K}}(\cdot)$, and $\langle \overline{\mathrm{E}} \rangle_{\mathcal{K}}(\cdot)$.

Proposition 7. *Given a $\mathcal{K}$-NFA $\mathcal{A}$ with n main states, one can construct, in polynomial time, $\mathcal{K}$-NFAs with $n+1$ main states accepting the languages $\langle \mathrm{B} \rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$, $\langle \mathrm{E} \rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$, $\langle \overline{\mathrm{B}} \rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$, and $\langle \overline{\mathrm{E}} \rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$, respectively.*

Proof. Let $\mathcal{A} = (S, M \times S, Q_0, \delta, F)$, where M is the set of main states.

Language $\langle \mathrm{B} \rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$. Let $\mathcal{A}_{\langle \mathrm{B} \rangle}$ be the NFA over S given by $\mathcal{A}_{\langle \mathrm{B} \rangle} = (S, (M \cup \{q_{acc}\}) \times S, Q_0, \delta', \{q_{acc}\} \times S)$, where $q_{acc} \notin M$ is a fresh main state, and for all $(q,s) \in (M \cup \{q_{acc}\}) \times S$ and $s' \in S$, we have $\delta'((q,s),s') = \emptyset$, if $s' \neq s$, and $\delta'((q,s),s)$ is defined as follows:

$$\delta'((q,s),s) = \begin{cases} \delta((q,s),s) & \text{if } (q,s) \in (M \times S) \setminus F \\ \delta((q,s),s) \cup (\{q_{acc}\} \times R(s)) & \text{if } (q,s) \in F \\ \{q_{acc}\} \times R(s) & \text{if } q = q_{acc}. \end{cases}$$

Given an input word ρ, from an initial state (q_0, s) of $\mathcal{A}$, the automaton $\mathcal{A}_{\langle \mathrm{B} \rangle}$ simulates the behavior of $\mathcal{A}$ from (q_0, s) over ρ. When $\mathcal{A}$ is in an accepting state (q_f, s) and the current input symbol is s, $\mathcal{A}_{\langle \mathrm{B} \rangle}$ can additionally choose to move to a state in $\{q_{acc}\} \times R(s)$, which is accepting for $\mathcal{A}_{\langle \mathrm{B} \rangle}$ (a prefix of ρ belongs to $\mathcal{L}(\mathcal{A})$). From such states, $\mathcal{A}_{\langle \mathrm{B} \rangle}$ accepts if and only if the remaining part of the input is a trace of $\mathcal{K}$. Since $\mathcal{A}$ is a $\mathcal{K}$-NFA, $\mathcal{A}_{\langle \mathrm{B} \rangle}$ is a $\mathcal{K}$-NFA by construction. Moreover, a word ρ over S is accepted by $\mathcal{A}_{\langle \mathrm{B} \rangle}$ *if and only if* ρ is a trace of $\mathcal{K}$ having some proper prefix ρ' in $\mathcal{L}(\mathcal{A})$ (note that $\rho' \neq \varepsilon$ since $\mathcal{A}$ is a $\mathcal{K}$-NFA). Thus, $\mathcal{L}(\mathcal{A}_{\langle \mathrm{B} \rangle}) = \langle \mathrm{B} \rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$.

Language $\langle \overline{\mathrm{B}} \rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$. Let $\mathcal{A}_{\langle \overline{\mathrm{B}} \rangle}$ be the NFA over S given by $\mathcal{A}_{\langle \overline{\mathrm{B}} \rangle} = (S, (M \cup \{q'_0\}) \times S, \{q'_0\} \times S, \delta', F')$, where $q'_0 \notin M$ is a fresh main state and δ' and F' are defined as follows: (i) for all $(q,s) \in (M \cup \{q'_0\}) \times S$ and $s' \in S$, we have $\delta'((q,s),s') = \emptyset$, if $s' \neq s$, and

$$\delta'((q,s),s) = \begin{cases} \displaystyle\bigcup_{(q_0,s) \in Q_0} \delta((q_0,s),s) & \text{if } q = q'_0 \\ \delta((q,s),s) & \text{otherwise.} \end{cases}$$

(ii) The set F' of accepting states is the set of states (q, s) of $\mathcal{A}$ such that there exists a run of $\mathcal{A}$ from (q, s) to some state in F over some non-empty word. Note that the set F' can be computed in time polynomial in the size of $\mathcal{A}$. By construction, we have that $\mathcal{A}_{\langle\overline{\mathrm{B}}\rangle}$ is a $\mathcal{K}$-NFA and $\mathcal{A}_{\langle\overline{\mathrm{B}}\rangle}$ accepts a word ρ *if and only if* ρ is a non-empty proper prefix of some word accepted by $\mathcal{A}$, implying that $\mathcal{L}(\mathcal{A}_{\langle\overline{\mathrm{B}}\rangle}) = \langle\overline{\mathrm{B}}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$.

The constructions for $\langle\mathrm{E}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$ and $\langle\overline{\mathrm{E}}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$—which are symmetric with respect to $\langle\mathrm{B}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$ and $\langle\overline{\mathrm{B}}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$—can be found in A.1. □

Now we show that $\mathcal{K}$-NFAs are closed under Boolean operations.

Proposition 8. *Given two $\mathcal{K}$-NFAs $\mathcal{A}$ and $\mathcal{A}'$ with n and n' main states, respectively, one can construct:*

- *a $\mathcal{K}$-NFA with $n + n'$ main states accepting $\mathcal{L}(\mathcal{A}) \cup \mathcal{L}(\mathcal{A}')$, in time $O(n + n')$;*
- *a $\mathcal{K}$-NFA with $2^{n+1} + 1$ main states accepting $\mathrm{Trc}_{\mathcal{K}} \setminus \mathcal{L}(\mathcal{A})$, in time $2^{O(n)}$.*

Proof. The construction for union is standard and thus omitted. The construction for complementation follows.

Let $\mathcal{A} = (S, M \times S, Q_0, \delta, F)$. First, we need a preliminary construction. Let us consider the NFA $\mathcal{A}'' = (S, (M \cup \{q_{acc}\}) \times S, Q_0, \delta'', \{q_{acc}\} \times S)$, where $q_{acc} \notin M$ is a fresh main state, and for all $(q, s) \in (M \cup \{q_{acc}\}) \times S$ and $s' \in s$, we have $\delta''((q, s), s') = \emptyset$, if $s' \neq s$, and

$$\delta''((q,s),s) = \begin{cases} \delta((q,s),s) \cup (\{q_{acc}\} \times S) & \text{if } q \in M \text{ and } \delta((q,s),s) \cap F \neq \emptyset \\ \delta((q,s),s) & \text{if } q \in M \text{ and } \delta((q,s),s) \cap F = \emptyset \\ \emptyset & \text{if } q = q_{acc}. \end{cases}$$

Note that $\mathcal{L}(\mathcal{A}'') = \mathcal{L}(\mathcal{A})$, but $\mathcal{A}''$ is actually *not* a $\mathcal{K}$-NFA.

Next, we show that it is possible to construct in time $2^{O(n)}$ a *weak* $\mathcal{K}$-NFA $\mathcal{A}_c$ with 2^{n+1} main states accepting $(\mathrm{Trc}_{\mathcal{K}} \setminus \mathcal{L}(\mathcal{A}'')) \cup \{\varepsilon\}$, where a *weak* $\mathcal{K}$-NFA is just a $\mathcal{K}$-NFA without the requirement that the empty word ε is not accepted. Thus, since a weak $\mathcal{K}$-NFA can be easily converted into an equivalent $\mathcal{K}$-NFA by using an additional main state, and $\mathcal{L}(\mathcal{A}'') = \mathcal{L}(\mathcal{A})$, the result follows. The weak $\mathcal{K}$-NFA $\mathcal{A}_c$ is given by $\mathcal{A}_c = (S, 2^{\tilde{M}} \times S, Q_{0,c}, \delta_c, F_c)$, where $\tilde{M} = M \cup \{q_{acc}\}$, and $Q_{0,c}$, F_c and δ_c are defined as follows:

- $Q_{0,c} = \{(P, s) \in 2^M \times S \mid P = \{q \in M \mid (q, s) \in Q_0\}\}$;
- $F_c = \{(P, s) \in 2^M \times S\}$;
- for all $(P, s) \in 2^{\tilde{M}} \times S$ and $s' \in S$, we have $\delta_c((P, s), s') = \emptyset$, if $s' \neq s$, and

$$\delta_c((P,s),s) = \bigcup_{s' \in R(s)} \Big\{(\{q' \in \tilde{M} \mid (q', s') \in \bigcup_{p \in P} \delta''(p, s)\}, s')\Big\}.$$

By construction, $\mathcal{A}_c$ is a weak $\mathcal{K}$-NFA not accepting words in $S^+ \setminus \mathrm{Trc}_{\mathcal{K}}$. Since $Q_{0,c} \subseteq F_c$, we have $\varepsilon \in \mathcal{L}(\mathcal{A}_c)$. Let $\rho \in \mathrm{Trc}_{\mathcal{K}}$ with $|\rho| = k$. To conclude the proof, we have to show that $\rho \in \mathcal{L}(\mathcal{A}'')$ if and only if $\rho \notin \mathcal{L}(\mathcal{A}_c)$.

Assuming that $\rho \in \mathcal{L}(\mathcal{A}'')$, we prove by contradiction that $\rho \notin \mathcal{L}(\mathcal{A}_c)$. Let us assume that there is a run of $\mathcal{A}_c$ over ρ having the form $(P_0, s_0) \cdots (P_k, s_k)$ such that $(P_0, s_0) \in Q_{0,c}$ and $(P_k, s_k) \in F_c$ implying that $q_{acc} \notin P_k$. By construction, $P_0 = \{q \in M \mid (q, s_0) \in Q_0\}$, and for all $i \in [0, k-1]$, $s_i = \rho(i)$ and $P_{i+1} = \{p \in \tilde{M} | (p, s_{i+1}) \in \delta''(q, s_i)$ for some $q \in P_i\}$. Since $\rho \in \mathcal{L}(\mathcal{A}'')$, there is $s \in S$, $(q_0, s_0) \in Q_0$ and an accepting run of $\mathcal{A}''$ over ρ having the form $(q_0, s_0) \cdots (q_{k-1}, s_{k-1})(q_k, s)$ where $q_k = q_{acc}$. By definition of the transition function δ'' of $\mathcal{A}''$, we can also assume that $s = s_k$. It follows that $q_i \in P_i$ for all $i \in [0, k]$, which is a contradiction since $q_{acc} \notin P_k$. Therefore $\rho \notin \mathcal{L}(\mathcal{A}_c)$.

As for the converse direction, let us assume that $\rho \notin \mathcal{L}(\mathcal{A}_c)$. We have to show that $\rho \in \mathcal{L}(\mathcal{A}'')$. By construction, there exists some run of $\mathcal{A}_c$ over ρ starting from an initial state (recall that $R(s) \neq \emptyset$ for all $s \in S$). Moreover, each of these runs has the form $(P_0, s_0) \cdots (P_k, s_k)$ such that $P_0 = \{q \in M \mid (q, s_0) \in Q_0\}$, $q_{acc} \in P_k$, and for all $i \in [0, k-1]$, $s_i = \rho(i)$ and $P_{i+1} = \{p \in \tilde{M} | (p, s_{i+1}) \in \delta(q, s_i)$ for some $q \in P_i\}$. It easily follows that there is an accepting run of $\mathcal{A}''$ over ρ from some initial state in $P_0 \times \{s_0\}$, thus proving the thesis. □

An MC algorithm for full HS can be built as follows. Let φ be an HS formula. First of all, we convert φ into an equivalent formula, called *existential form of* φ, that makes use of negations, disjunctions, and the existential modalities $\langle\mathrm{B}\rangle$, $\langle\overline{\mathrm{B}}\rangle$, $\langle\mathrm{E}\rangle$, and $\langle\overline{\mathrm{E}}\rangle$, only. For all $h \geq 1$, let HS_h denote the syntactical HS fragment consisting only of formulas φ such that the *nesting depth of negation in the existential form of* φ *is at most* h. Moreover $\neg\mathsf{HS}_h$ is the set of formulas φ such that $\neg\varphi \in \mathsf{HS}_h$.

Given an HS formula φ, checking whether $\mathcal{K} \not\models \varphi$ reduces to checking the existence of an initial trace ρ of $\mathcal{K}$ such that $\mathcal{K}, \rho \models \neg\varphi$. By exploiting Proposition 6, 7, and 8, we can build in a compositional way (driven by the structure of $\neg\varphi$) a $\mathcal{K}$-NFA $\mathcal{A}$ accepting the set of initial traces ρ such that $\mathcal{K}, \rho \models \neg\varphi$ and check $\mathcal{A}$ for emptiness. The next theorem states the main result of the section.

Theorem 9. *There exists a constant c such that, given a finite Kripke structure $\mathcal{K}$ and an HS formula φ, one can construct a $\mathcal{K}$-NFA with $O(|\mathcal{K}| \cdot \mathsf{Tower}(h, |\varphi|^c))$ states accepting the set of traces ρ of $\mathcal{K}$ such that $\mathcal{K}, \rho \models \varphi$, where h is the nesting depth of negation in the existential form of φ.*

*Moreover, for each $h \geq 0$, the MC problem for $\neg\mathsf{HS}_h$ is in h-***EXPTIME***. Additionally, for a constant-length formula, the MC problem is in* **P**.

It is worth recalling a result, proved in [6] for HS under the hypothesis of homogeneity, which immediately propagates to the complexity of the MC problem for full HS extended with regular expressions.

Theorem 10. *The MC problem for HS formulas over finite Kripke structures is* **EXPSPACE***-hard (under polynomial-time reductions).*

In the next section, we focus on the complexity of fragments of HS.

5. The fragments $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}E\overline{B}\overline{E}}$

In this section we focus on the syntactically maximal fragments $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}E\overline{B}\overline{E}}$ of HS showing that they feature a lower computational complexity w.r.t. the general case. First of all we prove in Section 5.1 that they enjoy an exponential small-model property, stating that if ρ is a trace of a finite Kripke structure $\mathcal{K}$ and ψ is an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula, then there is a trace ρ' such that $\mathcal{K}, \rho \models \psi$ if and only if $\mathcal{K}, \rho' \models \psi$ and $|\rho'|$ is exponential in the nesting depth of the $\langle \mathrm{B} \rangle$ modality in ψ.

In Section 5.2, we exploit this small-model property to design a MC algorithm for $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ belonging to the complexity class $\mathbf{AEXP_{pol}}$, namely, the class of problems decidable by singly exponential-time bounded Alternating Turing Machines with a polynomial-bounded number of alternations. Finally, in Section 5.3, we show that MC for $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ (actually the smaller fragment $\mathsf{B\overline{E}}$ would suffice) is hard for $\mathbf{AEXP_{pol}}$, hence proving the completeness for that class.

5.1. Exponential small-model property for $\mathsf{A\overline{A}B\overline{B}\overline{E}}$

Here we prove the *exponential small-model property* for $\mathsf{A\overline{A}B\overline{B}\overline{E}}$, which will be used as the basic step to prove that the MC problem for $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ belongs to $\mathbf{AEXP_{pol}}$.

Let us consider a finite Kripke structure $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ and a finite set $\mathsf{spec} = \{r_1, \ldots, r_H\}$ of (propositional-based) regular expressions over $\mathcal{AP}$. The small-model ensures that for each $h \geq 0$ and trace ρ of $\mathcal{K}$, it is possible to build another trace ρ' of $\mathcal{K}$, of bounded exponential length, which is indistinguishable from ρ with respect to the fulfillment of any $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ having atomic formulas in spec and nesting depth of the modality $\langle \mathrm{B} \rangle$ at most h (written $\mathrm{d_B}(\varphi) \leq h$). Formally, $\mathrm{d_B}(\varphi)$ is inductively defined as follows:

- $\mathrm{d_B}(r) = 0$, for any RE r over $\mathcal{AP}$;
- $\mathrm{d_B}(\neg\psi) = \mathrm{d_B}(\psi)$;
- $\mathrm{d_B}(\psi \wedge \phi) = \max\{\mathrm{d_B}(\psi), \mathrm{d_B}(\phi)\}$;
- $\mathrm{d_B}(\langle \mathrm{B} \rangle\, \psi) = 1 + \mathrm{d_B}(\psi)$;
- $\mathrm{d_B}(\langle \mathrm{X} \rangle\, \psi) = \mathrm{d_B}(\psi)$, for $X \in \{A, \overline{A}, \overline{B}, \overline{E}\}$.

In order to state the result, we first introduce the notion of *h-prefix bisimilarity* between a pair of traces ρ and ρ' of $\mathcal{K}$. As proved by Proposition 16 below, h-prefix bisimilarity is a sufficient condition for two traces ρ and ρ' to be indistinguishable with respect to the fulfillment of any $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ over spec with $\mathrm{d_B}(\varphi) \leq h$. Then, for a given trace ρ, we show how to determine a subset of positions of ρ, called the *h-prefix sampling* of ρ, that allows us to build another trace ρ' with single exponential length (both in h and $|\mathsf{spec}|$, where $|\mathsf{spec}|$ is defined as $\sum_{r \in \mathsf{spec}} |r|$) such that ρ and ρ' are h-prefix bisimilar.

For a regular expression r_ℓ in spec, with $\ell \in [1, H]$, let $\mathcal{A}_\ell = (2^{\mathcal{AP}}, Q_\ell, Q_\ell^0, \Delta_\ell, F_\ell)$ be the *canonical (complete)* NFA accepting $\mathcal{L}(r_\ell)$ (recall that $|Q_\ell| \leq 2|r_\ell|$). Without loss of generality, we assume that the sets of states of these automata are pairwise disjoint.

The notion of prefix bisimilarity uses the notion of *summary* of a trace ρ of $\mathcal{K}$, namely a tuple "recording" the initial and final states of ρ, and, for each automaton $\mathcal{A}_\ell$, with $\ell \in [1, H]$, the pairs of states $q, q' \in Q_\ell$ such that some run of $\mathcal{A}_\ell$ over $\mu(\rho)$ takes from q to q'.

Definition 11 *(Summary of a trace).* Let ρ be a trace of $\mathcal{K}$ with $|\rho| = n$. The summary $\mathcal{S}(\rho)$ of ρ (w.r.t. spec) is the triple $(\rho(1), \Pi, \rho(n))$, where

$$\Pi = \{(q, q') \mid q, q' \in Q_\ell \text{ for some } \ell \in [1, H], \text{and there is a run of } \mathcal{A}_\ell \text{ from } q \text{ to } q' \text{ over } \mu(\rho)\}.$$

Note that the number of summaries is at most $|S|^2 \cdot 2^{(2|\mathsf{spec}|)^2}$. The following result can be easily proved.

Proposition 12. *Let $h \geq 0$, and ρ and ρ' be two traces of $\mathcal{K}$ such that $\mathcal{S}(\rho) = \mathcal{S}(\rho')$. Then, for all regular expressions $r \in$ spec and traces ρ_L and ρ_R of $\mathcal{K}$ such that $\rho_L \star \rho$ and $\rho \star \rho_R$ are defined, the following properties hold:*

1. *$\mu(\rho) \in \mathcal{L}(r)$ if and only if $\mu(\rho') \in \mathcal{L}(r)$;*
2. *$\mathcal{S}(\rho_L \star \rho) = \mathcal{S}(\rho_L \star \rho')$;*
3. *$\mathcal{S}(\rho \star \rho_R) = \mathcal{S}(\rho' \star \rho_R)$.*

We now introduce the notion of *prefix bisimilarity* between a pair of traces ρ and ρ' of $\mathcal{K}$.

Definition 13 *(Prefix bisimilarity).* Let $h \geq 0$. Two traces ρ and ρ' of $\mathcal{K}$ are *h-prefix bisimilar* (w.r.t. spec) if the following conditions inductively hold:

- for $h = 0$: $\mathcal{S}(\rho) = \mathcal{S}(\rho')$;
- for $h > 0$: $\mathcal{S}(\rho) = \mathcal{S}(\rho')$ and for each proper prefix ν of ρ (respectively, proper prefix ν' of ρ'), there exists a proper prefix ν' of ρ' (respectively, proper prefix ν of ρ) such that ν and ν' are $(h-1)$-prefix bisimilar.

Property 14. For all $h \geq 0$, h-prefix bisimilarity is an equivalence relation over traces of $\mathcal{K}$.

The h-prefix bisimilarity of two traces ρ and ρ' is preserved by right (respectively, left) $\star$-concatenation with another trace of $\mathcal{K}$:

Proposition 15. *Let $h \geq 0$, and ρ and ρ' be two h-prefix bisimilar traces of $\mathcal{K}$. Then, for all traces ρ_L and ρ_R of $\mathcal{K}$ such that $\rho_L \star \rho$ and $\rho \star \rho_R$ are defined, the following holds:*

1. *$\rho_L \star \rho$ and $\rho_L \star \rho'$ are h-prefix bisimilar;*
2. *$\rho \star \rho_R$ and $\rho' \star \rho_R$ are h-prefix bisimilar.*

Proof. Let us note first that, since $\mathcal{S}(\rho) = \mathcal{S}(\rho')$, we have $\text{fst}(\rho) = \text{fst}(\rho')$ and $\text{lst}(\rho) = \text{lst}(\rho')$. Hence $\rho_L \star \rho$ (respectively, $\rho \star \rho_R$) is defined if and only if $\rho_L \star \rho'$ (respectively, $\rho' \star \rho_R$) is defined. The proofs of points (1.) and (2.) are by induction on $h \geq 0$.

(1.) Since ρ and ρ' are h-prefix bisimilar, $\mathcal{S}(\rho) = \mathcal{S}(\rho')$. By Proposition 12, $\mathcal{S}(\rho_L \star \rho) = \mathcal{S}(\rho_L \star \rho')$. Thus, if $h = 0$ (base case), the thesis follows. Now let $h > 0$ (induction step). Let us assume that ν is a proper prefix of $\rho_L \star \rho$ (the symmetric case, where we consider a proper prefix of $\rho_L \star \rho'$ is similar). We need to show that there exists a proper prefix ν' of $\rho_L \star \rho'$ such that ν and ν' are $(h-1)$-prefix bisimilar. If ν is a prefix of ρ_L, then we set $\nu' = \nu$ and the result trivially follows (note that, since ρ and ρ' are h-prefix bisimilar, it holds that $|\rho| > 1$ if and only if $|\rho'| > 1$). Otherwise, there is a proper prefix ξ of ρ such that $\nu = \rho_L \star \xi$. Since ρ and ρ' are h-prefix bisimilar, there exists a proper prefix ξ' of ρ' such that ξ and ξ' are $(h-1)$-prefix bisimilar. Thus, by setting $\nu' = \rho_L \star \xi'$, by the inductive hypothesis the thesis follows.

(2.) By Proposition 12, $\mathcal{S}(\rho \star \rho_R) = \mathcal{S}(\rho' \star \rho_R)$. Thus, if $h = 0$, the thesis follows. Now, let us assume that $h > 0$. We proceed by a double induction on $|\rho_R|$. As for the base case, where $|\rho_R| = 1$, the result is obvious. Thus let us assume that $|\rho_R| > 1$. Let ν be a proper prefix of $\rho \star \rho_R$ (the symmetric case, where we consider a proper prefix of $\rho' \star \rho_R$ is similar). We need to show that there exists a proper prefix ν' of $\rho' \star \rho_R$ such that ν and ν' are $(h-1)$-prefix bisimilar. If $\nu = \rho$ or ν is a proper prefix of ρ, then there exists a prefix ν' of ρ' such that ν and ν' are $(h-1)$-prefix bisimilar. Thus, since ν' is a proper prefix of $\rho' \star \rho_R$, the result follows. Otherwise, there exists a proper prefix ξ of ρ_R such that $\nu = \rho \star \xi$. By setting $\nu' = \rho' \star \xi$, and considering the inductive hypothesis on $|\rho_R|$, we obtain that ν and ν' are h-prefix bisimilar, hence $(h-1)$-prefix bisimilar as well, concluding the proof. □

By exploiting Propositions 12 and 15, we can prove that h-prefix bisimilarity preserves the fulfillment of $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formulas over spec having nesting depth of modality $\langle \mathrm{B} \rangle$ at most h.

Proposition 16. *Let $h \geq 0$, and ρ and ρ' be two h-prefix bisimilar traces of $\mathcal{K}$. Then, for each $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula ψ over spec with $\mathrm{d_B}(\psi) \leq h$, we have:*

$$\mathcal{K}, \rho \models \psi \iff \mathcal{K}, \rho' \models \psi.$$

Proof. We prove the proposition by a nested induction on the structure of the formula ψ and on the nesting depth $\mathrm{d_B}(\psi)$.

As for the base case, ψ is a regular expression in spec. Since $\mathcal{S}(\rho) = \mathcal{S}(\rho')$ (as ρ and ρ' are h-prefix bisimilar) the thesis holds by Proposition 12.

Let us now consider the inductive step. The cases where the root modality of ψ is a Boolean connective directly follow by the inductive hypothesis. As for the cases where the root modality is $\langle \mathrm{A} \rangle$ or $\langle \overline{\mathrm{A}} \rangle$, the result follows from the fact that, being ρ and ρ' h-prefix bisimilar, we have $\text{fst}(\rho) = \text{fst}(\rho')$ and $\text{lst}(\rho) = \text{lst}(\rho')$. It remains to consider the cases where the

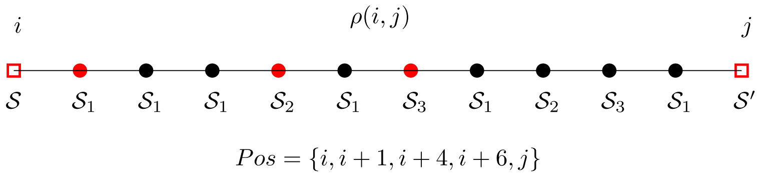


Fig. 4. Example of prefix-skeleton sampling Pos of a trace ρ in the interval $[i, j]$.

root modality is $\langle B\rangle$, $\langle \overline{B}\rangle$, or $\langle \overline{E}\rangle$. We prove the implication $\mathcal{K}, \rho \models \psi \implies \mathcal{K}, \rho' \models \psi$ (the converse implication is similar). Let $\mathcal{K}, \rho \models \psi$.

- $\psi = \langle B\rangle\, \varphi$: since $0 < \mathrm{d_B}(\psi) \le h$, it holds that $h > 0$. As $\mathcal{K}, \rho \models \langle B\rangle\, \varphi$, there is a proper prefix ν of ρ such that $\mathcal{K}, \nu \models \varphi$. Since ρ and ρ' are h-prefix bisimilar, there is a proper prefix ν' of ρ' such that ν and ν' are $(h-1)$-prefix bisimilar. Being $\mathrm{d_B}(\varphi) \le h-1$, by the inductive hypothesis we obtain that $\mathcal{K}, \nu' \models \varphi$. Hence $\mathcal{K}, \rho' \models \langle B\rangle\, \varphi$.
- $\psi = \langle \overline{B}\rangle\, \varphi$: since $\mathcal{K}, \rho \models \langle \overline{B}\rangle\, \varphi$, there is a trace ρ_R such that $|\rho_R| > 1$ and $\mathcal{K}, \rho \star \rho_R \models \varphi$. By Proposition 15, $\rho \star \rho_R$ and $\rho' \star \rho_R$ are h-prefix bisimilar. By the inductive hypothesis on the structure of the formula, we obtain that $\mathcal{K}, \rho' \star \rho_R \models \varphi$, hence, $\mathcal{K}, \rho' \models \langle \overline{B}\rangle\, \varphi$.
- $\psi = \langle \overline{E}\rangle\, \varphi$: this case is similar to the previous one. □

In the following, we show how a trace ρ, whose length exceeds a suitable exponential bound—precisely, $(|S| \cdot 2^{(2|\mathrm{spec}|)^2})^{h+2}$—can be contracted preserving h-prefix bisimilarity and, consequently, fulfillment of formulas φ with $\mathrm{d_B}(\varphi) \le h$. The basic contraction step of ρ is performed by choosing a subset of ρ-positions called *h-prefix sampling* (PS_h). A contraction can be performed whenever there are two positions $\ell < \ell'$ satisfying $\mathcal{S}(\rho(1,\ell)) = \mathcal{S}(\rho(1,\ell'))$ in between two consecutive positions in the linear ordering of PS_h. We prove that by taking the contraction $\rho' = \rho(1,\ell) \cdot \rho(\ell'+1, |\rho|)$, we obtain a trace of $\mathcal{K}$ which is h-prefix bisimilar to ρ. The basic contraction step can then be iterated over ρ' until the length bound is reached.

The notion of h-prefix sampling is inductively defined using the definition of *prefix-skeleton sampling*. For a set I of natural numbers, by “two consecutive elements of I” we mean a pair of elements $i, j \in I$ such that $i < j$ and $I \cap [i, j] = \{i, j\}$.

Definition 17 *(Prefix-skeleton sampling).* Let ρ be a trace of $\mathcal{K}$. Given two ρ-positions i and j, with $i \le j$, the *prefix-skeleton sampling of ρ in the interval* $[i, j]$ is the *minimal* set $Pos \supseteq \{i, j\}$ of ρ-positions in the interval $[i, j]$ satisfying the following condition:

- for each $k \in [i+1, j-1]$, the minimal position $k' \in [i+1, j-1]$ such that $\mathcal{S}(\rho(1,k')) = \mathcal{S}(\rho(1,k))$ belongs to Pos.

An example of prefix-skeleton sampling Pos of a trace ρ in the interval $[i, j]$ is given in Fig. 4. Assuming that $\mathcal{S}(\rho(1,u)) = \mathcal{S}_1$ for $u \in \{i+1, i+2, i+3, i+5, i+7, i+10\}$, $\mathcal{S}(\rho(1,u')) = \mathcal{S}_2$ for $u' \in \{i+4, i+8\}$, and $\mathcal{S}(\rho(1,u'')) = \mathcal{S}_3$ for $u'' \in \{i+6, i+9\}$, we have that $Pos = \{i, i+1, i+4, i+6, j\}$.

Notice that, as an immediate consequence of Definition 17, the prefix-skeleton sampling Pos of (any) trace ρ in an interval $[i, j]$ of ρ-positions is such that $|Pos| \le (|S| \cdot 2^{(2|\mathrm{spec}|)^2}) + 2$.

Definition 18 *(h-prefix sampling).* Let $h \ge 0$. The *h-prefix sampling of a trace* ρ of $\mathcal{K}$ is the *minimal* set PS_h of ρ-positions inductively satisfying the following conditions:

- Base case: $h = 0$. $PS_0 = \{1, |\rho|\}$;
- Inductive step: $h > 0$. (i) $PS_h \supseteq PS_{h-1}$ and (ii) for all pairs of consecutive positions i, j in PS_{h-1}, the prefix-skeleton sampling of ρ in the interval $[i, j]$ belongs to PS_h.

Let $i_1 < \ldots < i_N$ be the ordered sequence of positions in PS_h (note that $i_1 = 1$ and $i_N = |\rho|$). The *h-sampling word of* ρ is the sequence of summaries $\mathcal{S}(\rho(1,i_1)) \cdots \mathcal{S}(\rho(1,i_N))$.

We can state the following upper bound to the cardinality of prefix samplings.

Property 19. The cardinality of the h-prefix sampling PS_h of a trace ρ of $\mathcal{K}$ is such that

$$|PS_h| \le (|S| \cdot 2^{(2|\mathrm{spec}|)^2})^{h+1}.$$

As proved in the following lemma, for two traces, the property of having the same h-sampling word, is a sufficient condition to guarantee that they are h-prefix bisimilar.

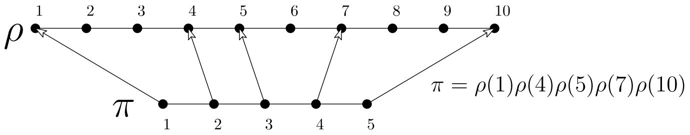


Fig. 5. A trace $\pi = \rho(1)\rho(4)\rho(5)\rho(7)\rho(10)$ induced by ρ.

Lemma 20. *For $h \geq 0$, two traces having the same h-sampling word are h-prefix bisimilar.*

Proof. The proof of the Lemma can be immediately derived by a stronger result stated in the following Claim.

Claim 21. Let $h \geq 0$, ρ and ρ' be two traces of $\mathcal{K}$, and PS_h and PS'_h be the two h-prefix samplings of ρ and ρ', respectively. Assume that ρ and ρ' have the same h-sampling word, namely there is $N \geq 1$ such that $PS_h : i_1 < i_2 < \ldots < i_N$, $PS'_h : i'_1 < i'_2 < \ldots < i'_N$, and for all $j \in [1, N]$, $\mathcal{S}(\rho(1, i_j)) = \mathcal{S}(\rho'(1, i'_j))$.

Then, for all $j \in [1, N-1]$, $n \in [i_j + 1, i_{j+1}]$ and $n' \in [i'_j + 1, i'_{j+1}]$ such that $\mathcal{S}(\rho[1, n]) = \mathcal{S}(\rho'[1, n'])$, it holds that $\rho(1, n)$ and $\rho'(1, n')$ are h-prefix bisimilar.

Proof. The proof is by induction on $h \geq 0$. For $h = 0$, the result is obvious. Now let us assume that $h > 0$. If $N = 1$ (respectively, $N = 2$), then $\rho = \rho'$ and $|\rho| = |\rho'| = N$, and the thesis trivially holds. Now, let us assume that $N > 2$. Since by hypothesis $\mathcal{S}(\rho(1, n)) = \mathcal{S}(\rho'(1, n'))$, we need to show that:

1. for each $m \in [1, n-1]$, there exists $m' \in [1, n'-1]$ such that $\rho(1, m)$ and $\rho'(1, m')$ are $(h-1)$-prefix bisimilar;
2. for each $m' \in [1, n'-1]$, there exists $m \in [1, n-1]$ such that $\rho(1, m)$ and $\rho'(1, m')$ are $(h-1)$-prefix bisimilar;

We only prove point (1.), the proof of (2.) being symmetric. We exploit in the proof the following fact that can be easily shown: let $k \in [0, h-1]$ and $1 = x_1 < \ldots < x_r = N$ be the subsequence of $1, \ldots, N$ such that $i_{x_1} < \ldots < i_{x_r}$ is the k-prefix sampling of ρ. Then, $i'_{x_1} < \ldots < i'_{x_r}$ is the k-prefix sampling of ρ'.

Now we prove (1.). Let $m \in [1, n-1]$. If $m = 1$, we set $m' = 1$, and the result follows. Now, let us assume that $m \geq 2$. Since $h > 0$, there must exist $x, y \in [1, N]$ such that $x < y$, $m \in [i_x + 1, i_y]$, and i_x and i_y are two consecutive positions in the $(h-1)$-prefix sampling of ρ. By the fact above, i'_x and i'_y are two consecutive positions in the $(h-1)$-prefix sampling of ρ'. We distinguish two cases:

- $m = i_y$. Since $n \in [i_j + 1, i_{j+1}]$ and $m < n$, it holds that $i_y \leq i_j$. Hence, $i'_y \leq i'_j$ as well. Moreover, since $n' > i'_j$, it holds that $i'_y < n'$. We set $m' = i'_y$. As $\mathcal{S}(\rho(1, i_y)) = \mathcal{S}(\rho'(1, i'_y))$, $m = i_y$, $m' = i'_y$, and i_x and i_y (respectively, i'_x and i'_y) are two consecutive positions in the $(h-1)$-prefix sampling of ρ (respectively, ρ'), the thesis follows by the inductive hypothesis on h.
- $m \neq i_y$. Hence, $m \in [i_x + 1, i_y - 1]$. Since i_x and i_y are two consecutive positions in the $(h-1)$-prefix sampling of ρ, there must exist $z \in [x+1, y-1]$ such that $i_z \leq m$ and $\mathcal{S}(\rho(1, m)) = \mathcal{S}(\rho(1, i_z))$. Since $i_z \leq m$, $m < n$, and $n \in [i_j + 1, i_{j+1}]$, it holds that $i_z \leq i_j$. Hence, $i'_z \leq i'_j < n'$. We set $m' = i'_z$. As $\mathcal{S}(\rho(1, i_z)) = \mathcal{S}(\rho'(1, i'_z))$, we obtain that $\mathcal{S}(\rho(1, m)) = \mathcal{S}(\rho'(1, m'))$, $m \in [i_x + 1, i_y]$ and $m' \in [i'_x + 1, i'_y]$. Thus, being i_x and i_y (respectively, i'_x and i'_y) two consecutive positions in the $(h-1)$-prefix sampling of ρ (respectively, ρ'), by the inductive hypothesis on h, the result follows. □

This concludes the proof of the Claim and of Lemma 20. □

The sufficient condition of Lemma 20 allows us to finally state the exponential small-model property for $\mathsf{A\overline{A}B\overline{B}E}$. In the proof of Theorem 23 below, it is shown how to derive from any trace ρ of $\mathcal{K}$, an h-prefix bisimilar trace ρ' *induced by* ρ (in the sense that ρ' is obtained by contracting ρ, namely, by concatenating subtraces of ρ in an ordered way, provided that ρ' is another trace of $\mathcal{K}$) such that $|\rho'| \leq (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$. By Proposition 16, ρ' is indistinguishable from ρ w.r.t. the fulfillment of any $\mathsf{A\overline{A}B\overline{B}E}$ formula φ over the set of atomic formulas in spec such that $\mathrm{d_B}(\varphi) \leq h$. We preliminarily define the notion of *induced trace*.

Definition 22 *(Induced trace).* Let $\rho \in \mathrm{Trc}_{\mathcal{K}}$ be a trace with $|\rho| = n$. A *trace induced by* ρ is a trace $\pi \in \mathrm{Trc}_{\mathcal{K}}$ such that there exists an increasing sequence of ρ-positions $i_1 < \ldots < i_k$, with $i_1 = 1$, $i_k = n$, and $\pi = \rho(i_1) \cdots \rho(i_k)$. Moreover, we say that the π-position j and the ρ-position i_j are *corresponding*.

Note that if π is induced by ρ, then $\mathrm{fst}(\pi) = \mathrm{fst}(\rho)$, $\mathrm{lst}(\pi) = \mathrm{lst}(\rho)$, and $|\pi| \leq |\rho|$. See Fig. 5 for an example.

Theorem 23 *(Exponential small-model property for* $\mathsf{A\overline{A}B\overline{B}\overline{E}}$*). Let* ρ *be a trace of* $\mathcal{K}$ *and* $h \geq 0$*. Then, there exists a trace* ρ' *induced by* ρ*, whose length is at most* $(|S| \cdot 2^{(2|spec|)^2})^{h+2}$*, which is* h*-prefix bisimilar to* ρ*. In particular, for every* $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ *formula* ψ *with atomic formulas in* spec *and such that* $\mathrm{d_B}(\psi) \leq h$*, it holds that* $\mathcal{K}, \rho \models \psi \iff \mathcal{K}, \rho' \models \psi$*.*

Proof. We show that if $|\rho| > (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$, then there exists a trace ρ' induced by ρ such that $|\rho'| < |\rho|$ and ρ and ρ' have the same h-sampling word.

Assume that $|\rho| > (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$. Let $PS_h : 1 = i_1 < \ldots < i_N = |\rho|$ be the h-prefix sampling of ρ. By Property 19, $|PS_h| \leq (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+1}$. Since the number of distinct summaries (w.r.t. spec) associated with the prefixes of ρ is at most $|S| \cdot 2^{(2|\mathsf{spec}|)^2}$, there must be two consecutive positions i_j and i_{j+1} in PS_h such that for some $\ell, \ell' \in [i_j + 1, i_{j+1} - 1]$ with $\ell < \ell'$, $\mathcal{S}(\rho(1, \ell)) = \mathcal{S}(\rho(1, \ell'))$. It easily follows that the sequence ρ' given by $\rho' := \rho(1, \ell) \cdot \rho(\ell' + 1, |\rho|)$ is a trace induced by ρ such that $|\rho'| < |\rho|$, and ρ and ρ' have the same h-sampling word. Now, by Lemma 20 ρ and ρ' are h-prefix bisimilar and by applying Proposition 16 we have that $\mathcal{K}, \rho \models \psi \iff \mathcal{K}, \rho' \models \psi$. Now, if $|\rho'| \leq (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$ the thesis holds, otherwise a sequence of contraction steps as shown above can be performed, until the length of the contracted trace fulfills the requirement. □

5.2. $\mathbf{AEXP_{pol}}$-*membership of MC for* $\mathsf{A\overline{A}B\overline{B}\overline{E}}$

In this section, taking advantage of the exponential small-model property proved in the previous section, we design a MC algorithm for $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formulas belonging to the complexity class $\mathbf{AEXP_{pol}}$. We recall that $\mathbf{AEXP_{pol}}$ is the class of problems solvable by singly exponential-time bounded Alternating Turing Machines (ATMs, for short) performing at most a polynomial-bounded number of alternations. More formally, an ATM $\mathcal{M}$ (we refer to [13] for standard syntax and semantics of ATMs) is *singly exponential-time bounded* if there exists an integer constant $c \geq 1$ such that, for each input α, any computation starting on α halts after at most $2^{|\alpha|^c}$ steps. The ATM $\mathcal{M}$ has a *polynomial-bounded number of alternations* if there exists an integer constant $c' \geq 1$ such that, for all inputs α and computations π starting from α, the number of alternations of existential and universal configurations along π is at most $|\alpha|^{c'}$.

In the sequel we restrict ourselves w.l.o.g. to $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formulas in *negation normal form* (abbreviated as NNF, and also known as *positive normal form*), i.e., formulas where negation is applied only to atomic formulas (regular expressions).[3] Any formula can be converted (in linear time) into an equivalent formula in NNF, having at most double length (by using De Morgan's laws and duality of HS modalities). For φ in NNF, the *dual* of φ, denoted as $\widetilde{\varphi}$, is defined as the NNF of $\neg\varphi$.

The complexity measure of an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ that we shall consider is the standard *alternation depth*, denoted by $\Upsilon(\varphi)$, between the existential $\langle X \rangle$ and universal modalities $[X]$ (and vice versa) occurring in the NNF of φ, *for* $X \in \{\overline{B}, \overline{E}\}$. Note that the definition does not consider the modalities associated with the Allen's relations in $\{A, \overline{A}, B\}$. Moreover, let FMC be the set of pairs $(\mathcal{K}, \varphi)$ consisting of a Kripke structure $\mathcal{K}$ and an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ such that $\mathcal{K} \models \varphi$ (i.e., $\mathcal{K}$ is a model of φ). The following theorem states the complexity upper bound.

Theorem 24. *One can construct a singly exponential-time bounded ATM accepting* FMC *whose number of alternations on an input* $(\mathcal{K}, \varphi)$ *is at most* $\Upsilon(\varphi) + 2$*.*

To prove the assertion of Theorem 24 we define a procedure in the remaining part of the section. Such a procedure can be easily translated into an ATM (the translation is omitted).

We start with some auxiliary notation. Let us fix a finite Kripke structure $\mathcal{K}$ with set of states S and an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ in NNF. Let $h = \mathrm{d_B}(\varphi)$, and spec be the set of regular expressions occurring in φ. A *certificate* of $(\mathcal{K}, \varphi)$ is a trace ρ of $\mathcal{K}$ whose length is less than $(|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$ (the bound for the exponential small-model property of Theorem 23). A $\overline{B}$*-witness* (respectively, $\overline{E}$*-witness*) of a certificate ρ for $(\mathcal{K}, \varphi)$, is a certificate ρ' of $(\mathcal{K}, \varphi)$ such that ρ' is h-prefix bisimilar to a trace having the form $\rho \star \rho''$ (respectively, $\rho'' \star \rho$) for some *certificate* ρ'' of $(\mathcal{K}, \varphi)$ with $|\rho''| > 1$. By $\mathsf{SD}(\varphi)$ we denote the set consisting of the subformulas ψ of φ and the *duals* $\widetilde{\psi}$. The results stated in Section 5.1 are used to prove the properties of certificates listed in the following proposition and exploited in the MC algorithm.

Proposition 25. *Let* $\mathcal{K}$ *be a finite Kripke structure,* φ *be an* $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ *formula in NNF, and* ρ *be a certificate for* $(\mathcal{K}, \varphi)$*. The following properties hold:*

1. *for each* $\langle X \rangle\psi \in SD(\varphi)$*, with* $X \in \{\overline{B}, \overline{E}\}$*, it holds* $\mathcal{K}, \rho \models \langle X \rangle\psi$ *if and only if there exists an* X*-witness* ρ' *of* ρ *for* $(\mathcal{K}, \varphi)$ *such that* $\mathcal{K}, \rho' \models \psi$*;*
2. *for each trace having the form* $\rho \star \rho'$ *(respectively,* $\rho' \star \rho$*) such that* ρ' *is a certificate for* $(\mathcal{K}, \varphi)$*, one can construct in time singly exponential in the size of* $(\mathcal{K}, \varphi)$*, a certificate* ρ'' *which is* h*-prefix bisimilar to* $\rho \star \rho'$ *(respectively,* $\rho' \star \rho$*), with* $h = \mathrm{d_B}(\varphi)$*.*

[3] Not to be confused with the *negation form* used in the previous section.

Proof. (1.) Let $\langle X\rangle\psi \in \mathsf{SD}(\varphi)$ with $X \in \{\overline{B}, \overline{E}\}$, $h = \mathrm{d_B}(\varphi)$, and ρ be a certificate for $(\mathcal{K}, \varphi)$. Let us assume that $X = \overline{E}$ (the case for $X = \overline{B}$ is similar).

First we assume that there exists an $\overline{E}$-witness ρ' of ρ for $(\mathcal{K}, \varphi)$ such that $\mathcal{K}, \rho' \models \psi$. Hence ρ' is h-prefix bisimilar to a trace having the form $\rho'' \star \rho$, with $|\rho''| > 1$. Since $\langle \overline{E}\rangle\psi \in \mathsf{SD}(\varphi)$, it holds that $\mathrm{d_B}(\langle \overline{E}\rangle\psi) \le h$. By Proposition 16 we have that $\mathcal{K}, \rho'' \star \rho \models \psi$ and, then, $\mathcal{K}, \rho \models \langle \overline{E}\rangle\psi$.

To prove the converse implication, we assume that $\mathcal{K}, \rho \models \langle \overline{E}\rangle\psi$. Then, there exists a trace having the form $\rho'' \star \rho$ with $|\rho''| > 1$ such that $\mathcal{K}, \rho'' \star \rho \models \psi$. By Theorem 23 there exists a certificate ν for $(\mathcal{K}, \varphi)$ which is h-prefix bisimilar to ρ''. By Proposition 15, $\nu \star \rho$ is h-prefix bisimilar to $\rho'' \star \rho$. By applying Proposition 16 we deduce that $\mathcal{K}, \nu \star \rho \models \psi$. By applying again Theorem 23, there exists a certificate ρ' for $(\mathcal{K}, \varphi)$ which is h-prefix bisimilar to $\nu \star \rho$ such that $\mathcal{K}, \rho' \models \psi$. Thus, since ρ' is an $\overline{E}$-witness of ρ for $(\mathcal{K}, \varphi)$, the property of point (1.) follows.

(2.) From the trace $\rho \star \rho'$ (respectively, $\rho' \star \rho$), where both ρ and ρ' are certificates for $(\mathcal{K}, \varphi)$, we first compute the h-prefix sampling of $\rho \star \rho'$ (respectively, $\rho' \star \rho$), where $h = \mathrm{d_B}(\varphi)$. Then, proceeding as in the proof of Theorem 23, we extract from $\rho \star \rho'$ (respectively, $\rho' \star \rho$) a trace which is h-prefix bisimilar to $\rho \star \rho'$ (respectively, $\rho' \star \rho$). Since the lengths of ρ and ρ' are singly exponential in the sizes of $(\mathcal{K}, \varphi)$, the property of point (2.) follows. □

Let $\mathsf{A\overline{A}}(\varphi)$ be the set of formulas in $\mathsf{SD}(\varphi)$ having the form $\langle X\rangle\psi'$ or $[X]\psi'$, with $X \in \{A, \overline{A}\}$. An $\mathsf{A\overline{A}}$-labeling *Lab* for $(\mathcal{K}, \varphi)$ is a mapping associating with each state s of $\mathcal{K}$ a maximally consistent set of subformulas of $\mathsf{A\overline{A}}(\varphi)$. More precisely, for all $s \in S$, $Lab(s)$ is such that for all $\psi, \widetilde{\psi} \in \mathsf{A\overline{A}}(\varphi)$, $Lab(s) \cap \{\psi, \widetilde{\psi}\}$ is a singleton. We say that *Lab* is *valid* if, for all states $s \in S$ and $\psi \in Lab(s)$, we have $\mathcal{K}, s \models \psi$ (we consider s as a length-1 trace). Note that if *Lab* is valid, then (*i*) for each trace ρ of $\mathcal{K}$ and $\langle \mathrm{A}\rangle\,\psi' \in \mathsf{A\overline{A}}(\varphi)$ (respectively, $\langle \overline{\mathrm{A}}\rangle\,\psi' \in \mathsf{A\overline{A}}(\varphi)$), it holds that $\mathcal{K}, \rho \models \langle \mathrm{A}\rangle\,\psi'$ (respectively, $\mathcal{K}, \rho \models \langle \overline{\mathrm{A}}\rangle\,\psi'$) if and only if $\langle \mathrm{A}\rangle\,\psi' \in Lab(\mathrm{lst}(\rho))$ (respectively, $\langle \overline{\mathrm{A}}\rangle\,\psi' \in Lab(\mathrm{fst}(\rho))$). Analogously, (*ii*) for each trace ρ of $\mathcal{K}$ and $[A]\psi' \in \mathsf{A\overline{A}}(\varphi)$ (respectively, $[\overline{A}]\psi' \in \mathsf{A\overline{A}}(\varphi)$), it holds that $\mathcal{K}, \rho \models [A]\psi'$ (respectively, $\mathcal{K}, \rho \models [\overline{A}]\psi'$) if and only if $[A]\psi' \in Lab(\mathrm{lst}(\rho))$ (respectively, $[\overline{A}]\psi' \in Lab(\mathrm{fst}(\rho))$).

Finally, a *well-formed set for* $(\mathcal{K}, \varphi)$ is a finite set $\mathcal{W}$ consisting of pairs (ψ, ρ) such that $\psi \in \mathsf{SD}(\varphi)$ and ρ is a certificate of $(\mathcal{K}, \varphi)$. $\mathcal{W}$ is said to be *universal* if each formula occurring in $\mathcal{W}$ has the form $[X]\psi$, with $X \in \{\overline{B}, \overline{E}\}$. The *dual* $\widetilde{\mathcal{W}}$ of $\mathcal{W}$ is the well-formed set obtained by replacing each pair $(\psi, \rho) \in \mathcal{W}$ with $(\widetilde{\psi}, \rho)$. A well-formed set $\mathcal{W}$ is *valid* if, for each $(\psi, \rho) \in \mathcal{W}$, it holds that $\mathcal{K}, \rho \models \psi$.

Algorithm 1 `check`$(\mathcal{K}, \varphi)$ [*$\mathcal{K}$: finite Kripke structure, φ: $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula in NNF*]

```
1: existentially choose an $\mathsf{A\overline{A}}$-labeling $Lab$ for $(\mathcal{K}, \varphi)$
2: for each state $s$ and $\psi \in Lab(s)$ do
3:    Case $\psi = \langle \mathrm{A}\rangle\,\psi'$ (respectively, $\psi = \langle \overline{\mathrm{A}}\rangle\,\psi'$)
4:       existentially choose a certificate $\rho$ with $\mathrm{fst}(\rho) = s$ (respectively, $\mathrm{lst}(\rho) = s$)
5:       checkTrue$_{(\mathcal{K},\varphi,Lab)}(\{(\psi', \rho)\})$
6:    case $\psi = [A]\psi'$ (respectively, $\psi = [\overline{A}]\psi'$)
7:       universally choose a certificate $\rho$ with $\mathrm{fst}(\rho) = s$ (respectively, $\mathrm{lst}(\rho) = s$)
8:       checkTrue$_{(\mathcal{K},\varphi,Lab)}(\{(\psi', \rho)\})$
9:    EndCase
10: universally choose a certificate $\rho$ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s_0$      ▷ $s_0$ is the initial state of $\mathcal{K}$
11: checkTrue$_{(\mathcal{K},\varphi,Lab)}(\{(\varphi, \rho)\})$
```

We can now introduce the procedure `check` reported in Algorithm 1 that defines the ATM required to prove the assertion of Theorem 24. The procedure `check` takes a pair $(\mathcal{K}, \varphi)$ as input and: (1) it guesses an $\mathsf{A\overline{A}}$-labeling *Lab* for $(\mathcal{K}, \varphi)$ (line 1); (2) it checks that *Lab* is valid (lines 2–9); (3) for every certificate ρ starting from the initial state, it verifies that $\mathcal{K}, \rho \models \varphi$ (lines 10–11). To perform steps (2)–(3), it exploits the auxiliary ATM procedure `checkTrue` reported in Algorithm 2). The procedure `checkTrue` takes as input a well-formed set $\mathcal{W}$ for $(\mathcal{K}, \varphi)$ and, assuming that the current $\mathsf{A\overline{A}}$-labeling *Lab* is valid, checks whether $\mathcal{W}$ is valid. For each pair $(\psi, \rho) \in \mathcal{W}$ such that ψ does not have the form $[X]\psi'$, with $X \in \{\overline{B}, \overline{E}\}$, `checkTrue` directly checks whether or not $\mathcal{K}, \rho \models \psi$ (lines 4–29). In order to allow a deterministic choice of the current element of the iteration (line 2), we assume that the set $\mathcal{W}$ is implemented as an ordered data structure. At each iteration of the while loop in `checkTrue`, the current pair $(\psi, \rho) \in \mathcal{W}$ is processed according to the semantics of HS, exploiting the guessed $\mathsf{A\overline{A}}$-labeling *Lab* for modalities $\langle \mathrm{A}\rangle$, $\langle \overline{\mathrm{A}}\rangle$, $[A]$ and $[\overline{A}]$ (lines 10–15) and $\langle \overline{\mathrm{E}}\rangle$-witnesses and $\langle \overline{\mathrm{B}}\rangle$-witnesses guaranteed by Proposition 25 for modalities $\langle \overline{\mathrm{E}}\rangle$ and $\langle \overline{\mathrm{B}}\rangle$ (lines 26–28). The processing is either deterministic or based on an existential choice, and the currently processed pair (ψ, ρ) is either removed from $\mathcal{W}$, or replaced with pairs (ψ', ρ') such that ψ' is a strict subformula of ψ (it is the case of boolean connectives and modalities $\langle \mathrm{B}\rangle$, $[B]$, $\langle \overline{\mathrm{E}}\rangle$ and $\langle \overline{\mathrm{B}}\rangle$, lines 21–28).

At the end of the while loop, the resulting well formed set $\mathcal{W}$ is either empty or universal. In the former case, the procedure accepts (lines 30–31). In the latter case, there is a switch in the current operation mode (line 33). For each element (ψ, ρ) in the dual of $\mathcal{W}$—note that the root modality of ψ is either $\langle \overline{\mathrm{E}}\rangle$ or $\langle \overline{\mathrm{B}}\rangle$—the auxiliary ATM procedure `checkFalse` (reported in Algorithm 7 of A.2) is invoked, that accepts the input $\{(\psi, \rho)\}$ if and only if $\mathcal{K}, \rho \not\models \psi$. The procedure `checkFalse` is the "dual" of `checkTrue`, as it is simply obtained from `checkTrue` by switching *accept*

Algorithm 2 $\texttt{checkTrue}_{(\mathcal{K},\varphi,Lab)}(\mathcal{W})$ [*$\mathcal{W}$: well-formed set, Lab: A$\overline{\text{A}}$-labeling for $(\mathcal{K},\varphi)$*]

```
while $\mathcal{W}$ is not universal do
  deterministically select $(\psi,\rho) \in \mathcal{W}$ such that $\psi$ does not have the form $[\overline{E}]\psi'$ and $[\overline{B}]\psi'$
  $\mathcal{W} \leftarrow \mathcal{W} \setminus \{(\psi,\rho)\}$
  Case $\psi = r$ with $r \in \mathsf{RE}$
    if $\rho \notin \mathcal{L}(r)$ then
      reject the input
  case $\psi = \neg r$ with $r \in \mathsf{RE}$
    if $\rho \in \mathcal{L}(r)$ then
      reject the input
  case $\psi = \langle \mathrm{A} \rangle \psi'$ or $\psi = [A]\psi'$
    if $\psi \notin Lab(\mathrm{lst}(\rho))$ then
      reject the input
  case $\psi = \langle \overline{\mathrm{A}} \rangle \psi'$ or $\psi = [\overline{A}]\psi'$
    if $\psi \notin Lab(\mathrm{fst}(\rho))$ then
      reject the input
  case $\psi = \psi_1 \vee \psi_2$
    existentially choose $i = 1, 2$
    $\mathcal{W} \leftarrow \mathcal{W} \cup \{(\psi_i,\rho)\}$
  case $\psi = \psi_1 \wedge \psi_2$
    $\mathcal{W} \leftarrow \mathcal{W} \cup \{(\psi_1,\rho),(\psi_2,\rho)\}$
  case $\psi = \langle \mathrm{B} \rangle \psi'$
    existentially choose $\rho' \in \mathrm{Pref}(\rho)$
    $\mathcal{W} \leftarrow \mathcal{W} \cup \{(\psi',\rho')\}$
  case $\psi = [B]\psi'$
    $\mathcal{W} \leftarrow \mathcal{W} \cup \{(\psi',\rho') \mid \rho' \in \mathrm{Pref}(\rho)\}$
  case $\psi = \langle X \rangle \psi'$ with $X \in \{\overline{E},\overline{B}\}$
    existentially choose an $X$-witness $\rho'$ of $\rho$ for $(\mathcal{K},\varphi)$
    $\mathcal{W} \leftarrow \mathcal{W} \cup \{(\psi',\rho')\}$
  EndCase
if $\mathcal{W} = \emptyset$ then
  accept the input
else
  universally choose $(\psi,\rho) \in \widetilde{\mathcal{W}}$
  $\texttt{checkFalse}_{(\mathcal{K},\varphi,Lab)}(\{(\psi,\rho)\})$
```

and *reject*, by switching existential choices and universal choices, and by converting the last call to `checkFalse` into `checkTrue`. Thus `checkFalse` accepts an input $\mathcal{W}$ if and only if $\mathcal{W}$ is *not* valid.

Notice that the number of alternations of the ATM `check` between existential and universal choices is clearly the number of switches between the calls to the procedures `checkTrue` and `checkFalse`, plus 2, i.e. $\Upsilon(\varphi)+2$.

The correctness of the procedure `check` and its complexity bound is stated in the following Proposition that immediately implies Theorem 24.

Proposition 26. *The ATM* `check` *is a singly exponential-time bounded ATM accepting* FMC, *whose number of alternations on input* $(\mathcal{K},\varphi)$ *is at most* $\Upsilon(\varphi)+2$.

The proof of Proposition 26 is given in details in A.3. The proof exploits the exponential small-model property for A$\overline{\text{A}}$B$\overline{\text{B}}\overline{\text{E}}$ (Theorem 23) which allows us to consider only certificates, which are single exponential in the size of the input $(\mathcal{K},\varphi)$, instead of traces of arbitrary length.

5.3. $\mathbf{AEXP_{pol}}$-*hardness of MC for* B$\overline{\text{E}}$

Now we conclude Section 5 by showing that the MC problem for the fragment B$\overline{\text{E}}$ is $\mathbf{AEXP_{pol}}$-hard (implying the $\mathbf{AEXP_{pol}}$-hardness of A$\overline{\text{A}}$B$\overline{\text{B}}\overline{\text{E}}$). The result is obtained by a polynomial-time reduction from a variant of the domino-tiling problem for grids with exponential-length rows and columns, called *alternating multi-tiling problem*.

An instance of this problem is a tuple $\mathcal{I} = (n, D, D_0, H, V, M, D_{acc})$, where: n is a positive *even* natural number encoded in unary; D is a non-empty finite set of *domino types*; $D_0 \subseteq D$ is a set of *initial domino types*; $H \subseteq D \times D$ and $V \subseteq D \times D$ are the *horizontal* and *vertical matching relations*, respectively; $M \subseteq D \times D$ is the *multi-tiling matching relation*; $D_{acc} \subseteq D$ is a set of *accepting domino types*.

A *tiling of* $\mathcal{I}$ is a map assigning a domino type to each cell of a $2^n \times 2^n$ squared grid coherently with the horizontal and vertical matching relations. Formally a tiling of $\mathcal{I}$ is a map $f : [0, 2^n-1] \times [0, 2^n-1] \to D$ such that:

- for all $i, j \in [0, 2^n-1] \times [0, 2^n-1]$ with $j < 2^n - 1$, $(f(i,j), f(i,j+1)) \in H$ (*row-adjacency requirement*);
- for all $i, j \in [0, 2^n-1] \times [0, 2^n-1]$ with $i < 2^n - 1$, $(f(i,j), f(i+1,j)) \in V$ (*column-adjacency requirement*).

The *initial condition* $Init(f) := f(0,0)f(0,1)\ldots f(0,2^n-1)$ of the tiling f is the content of the first row of f. A *multi-tiling of* $\mathcal{I}$ is a tuple $(f_1,\ldots,f_n)$ of n tilings which are coherent w.r.t. the multi-tiling matching relation M, that is, such that:

- for all $i, j \in [0, 2^n-1] \times [0, 2^n-1]$ and $\ell \in [1, n-1]$, $(f_\ell(i,j), f_{\ell+1}(i,j)) \in M$ (*multi-cell requirement*), and
- $f_n(2^n-1, j) \in D_{acc}$ for some $j \in [0, 2^n-1]$ (*acceptance requirement*).

The *alternating multi-tiling problem* for an instance $\mathcal{I}$ is checking whether

$$\forall w_1 \in (D_0)^{2^n}, \exists w_2 \in (D_0)^{2^n}, \ldots, \forall w_{n-1} \in (D_0)^{2^n}, \exists w_n \in (D_0)^{2^n}$$

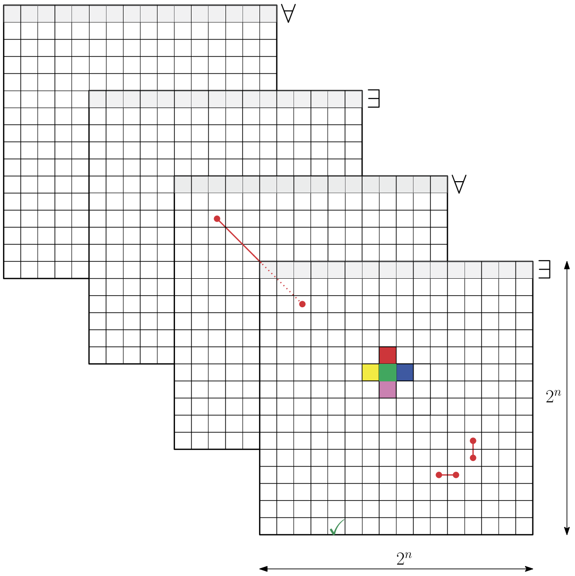


Fig. 6. The alternating multi-tiling problem (for $n = 4$). The red lines represent the row-adjacency, column-adjacency, and multi-cell requirements. The green tick denotes the acceptance requirement. The quantifiers ∀/∃ associated with the first rows of each tiling mean that the content of these rows have to be universally (respectively, existentially) selected, if they belong to an odd (respectively, even) tiling.

such that there exists a multi-tiling $(f_1, \ldots, f_n)$, where for all $i \in [1, n]$, $\mathit{Init}(f_i) = w_i$. See Fig. 6 for a visual representation of the alternating multi-tiling problem.

The following complexity result holds (for a proof we refer to [5]).

Theorem 27. *The alternating multi-tiling problem is* $\mathbf{AEXP_{pol}}$*-complete.*

The fact that the MC problem for the fragment $\mathsf{B\overline{E}}$ is $\mathbf{AEXP_{pol}}$-hard is an immediate corollary of the following result.

Theorem 28. *One can construct, in time polynomial in the size of $\mathcal{I}$, a finite Kripke structure $\mathcal{K}_\mathcal{I}$ and a $\mathsf{B\overline{E}}$ formula $\varphi_\mathcal{I}$ over the set of proposition letters $\mathcal{AP} = D \cup (\{r, c\} \times \{0, 1\}) \cup \{\bot, \mathit{end}\}$ such that $\mathcal{K}_\mathcal{I} \models \varphi_\mathcal{I}$ if and only if $\mathcal{I}$ is a* positive *instance of the alternating multi-tiling problem.*

The rest of this section is devoted to the construction of the Kripke structure $\mathcal{K}_\mathcal{I}$ and the $\mathsf{B\overline{E}}$ formula $\varphi_\mathcal{I}$ proving Theorem 28. Let $\mathcal{AP}$ be $D \cup (\{r, c\} \times \{0, 1\}) \cup \{\bot, \mathit{end}\}$. The Kripke structure $\mathcal{K}_\mathcal{I}$ is given by $\mathcal{K}_\mathcal{I} = (\mathcal{AP}, S, R, \mu, s_0)$, where $S = \mathcal{AP}$, $s_0 = \mathit{end}$, μ is the identity mapping (we identify a singleton set $\{p\}$ with p), and $R = \{(s, s') \mid s \in \mathcal{AP} \setminus \{\mathit{end}\}, s' \in \mathcal{AP}\}$. Note that the initial state *end* has no successors,[4] and that a trace of $\mathcal{K}_\mathcal{I}$ can be identified with its induced labeling sequence.

The construction of the $\mathsf{B\overline{E}}$ formula $\varphi_\mathcal{I}$ is based on a suitable encoding of multi-tilings which is described in the following. The symbols $\{r\} \times \{0, 1\}$ and $\{c\} \times \{0, 1\}$ in $\mathcal{AP}$ are used to encode the values of two n-bits counters numbering the 2^n rows and columns, respectively, of a tiling. For a multi-tiling $F = (f_1, \ldots, f_n)$ and for all $i, j \in [0, 2^n - 1]$, the (i, j)-th *multi-cell* $(f_1(i, j), \ldots, f_n(i, j))$ of F is encoded by the word C of length $3n$ over $\mathcal{AP}$, called *multi-cell code*, given by

$$d_1 \cdots d_n (r, b_1) \cdots (r, b_n)(c, b'_1) \cdots (c, b'_n),$$

where $b_1 \cdots b_n$ and $b'_1 \cdots b'_n$ are the binary encodings of the row number i and column number j, respectively, and for all $\ell \in [1, n]$, $d_\ell = f_\ell(i, j)$ (i.e., the content of the (i, j)-th cell of component f_ℓ). The *content* of C is $d_1 \cdots d_n$. Since F is a multi-tiling, the following well-formedness requirement must be satisfied by the encoding C: for all $\ell \in [1, n - 1]$, $(d_\ell, d_{\ell+1}) \in M$. We call such words *well-formed multi-cell codes*.

Definition 29 *(Multi-tiling codes).* A *multi-tiling code* is a finite word w over $\mathcal{AP}$ obtained by concatenating well-formed multi-cell codes so that the following conditions hold:

- for all $i, j \in [0, 2^n - 1]$, there is a multi-cell code in w with row number i and column number j *(completeness requirement)*;

[4] This violates Definition 1, but we define the state *end* to have no successors only for convenience.

- for all multi-cell codes C and C' occurring in w, if C and C' have the same row number and column number, then C and C' have the same content *(uniqueness requirement)*;
- for all multi-cell codes C and C' in w having the same row-number (respectively, column number), column numbers (respectively, row numbers) j and $j+1$, respectively, and contents $d_1 \cdots d_n$ and $d'_1 \cdots d'_n$, respectively, it holds that $(d_\ell, d'_\ell) \in H$ (respectively $(d_\ell, d'_\ell) \in V$) for all $\ell \in [1, n]$ *(row-adjacency requirement)* (respectively, *(column-adjacency requirement)*);
- there is a multi-cell code in w with row-number $2^n - 1$ whose content is in $D^{n-1} \cdot d_{acc}$ for some $d_{acc} \in D_{acc}$ *(acceptance requirement)*.

Finally, we encode the initial conditions of the components of a multi-tiling. An *initial cell code* encodes a cell of the first row of a tiling and is a word w of length $n+1$ having the form $w = d(c, b_1) \cdots (c, b_n)$, where $d \in D_0$ and $b_1, \ldots, b_n \in \{0, 1\}$. We say that d is the *content* of w and the integer in $[0, 2^n - 1]$ encoded by $b_1 \cdots b_n$ is the *column number* of w.

Definition 30 *(Multi-initialization codes).* An *initialization code* is a finite word w over $\mathcal{AP}$ which is the concatenation of initial cell codes such that:

- for all $i \in [0, 2^n - 1]$, there is an initial cell code in w with column number i.
- for all initial cell codes C and C' occurring in w, if C and C' have the same column number, then C and C' have the same content.

A *multi-initialization code* is a finite word over $\mathcal{AP}$ having the form $\bot \cdot w_n \cdots \bot \cdot w_1 \cdot end$ such that for all $\ell \in [1, n]$, w_ℓ is an initialization code.

Definition 31 *(Initialized multi-tiling codes).* An *initialized multi-tiling code* is a finite word over $\mathcal{AP}$ having the form $\bot \cdot w \cdot \bot \cdot w_n \cdots \bot \cdot w_1 \cdot end$ such that w is a multi-tiling code, $\bot \cdot w_n \cdots \bot \cdot w_1 \cdot end$ is a multi-initialization code, and the following requirement holds:

- for each multi-cell code in w having row number 0, column number i, and content $d_1 \cdots d_n$ and for all $\ell \in [1, n]$, there is an initial cell code in w_ℓ having column number i and content d_ℓ *(initialization coherence requirement)*.

Before proving Theorem 28, we sketch the idea for the construction of the $\mathsf{B}\overline{\mathsf{E}}$ formula $\varphi_{\mathcal{I}}$ ensuring that $\mathcal{K}_{\mathcal{I}} \models \varphi_{\mathcal{I}}$ if and only if $\mathcal{I}$ is a positive instance of the alternating multi-tiling problem. We preliminarily observe that since the initial state of $\mathcal{K}_{\mathcal{I}}$ has no successors, the only initial trace of $\mathcal{K}_{\mathcal{I}}$ is the trace *end* having length 1. To guess a trace corresponding to an initialized multi-tiling code, $\mathcal{K}_{\mathcal{I}}$ is unraveled backward starting from *end*, exploiting the modality $\overline{\mathsf{E}}$. The structure of the formula $\varphi_{\mathcal{I}}$ is

$$\varphi_{\mathcal{I}} := [\overline{E}](\varphi_1 \rightarrow \langle\overline{\mathsf{E}}\rangle(\varphi_2 \wedge (\ldots([\overline{E}](\varphi_{n-1} \rightarrow \langle\overline{\mathsf{E}}\rangle(\varphi_n \wedge \langle\overline{\mathsf{E}}\rangle\, \varphi_{IMT})))\ldots))).$$

It features $n+1$ unraveling steps starting from the initial trace *end*. The first n steps are used to guess a sequence of n initialization codes. Intuitively, each formula φ_i is used to constrain the i-th unraveling to be an initialization code, in such a way that at depth n in the formula a multi-initialization code is under evaluation. The last unraveling step (the innermost in the formula) is used to guess the multi-tiling code. The innermost formula φ_{IMT} is evaluated over a trace corresponding to an initialized multi-tiling code, and checks its structure: multi-cell codes are "captured" by regular expressions (encoding in particular their row and column numbers and contents). The completeness, uniqueness, row- and column-adjacency requirements for the multitiling of Definition 29 are enforced by the combined use of the $[\overline{E}]$ modality and regular expressions. The intuition of the technique is graphically depicted in Fig. 7, where w is a multi-tiling code. Since the problem is to check constraints between pairs of multi-cell codes occurring in arbitrary positions of w, we use the following trick. A copy of two multi-cell codes C and C' (see Fig. 7) are generated next to each other, as backward extensions of the initialized multi-tiling code, by means of modality $[\overline{E}]$. We then check that both C and C' occur in (arbitrary positions of) w, and, if it is the case, the required constraint is checked against the generated copies C and C', taking advantage of their adjacency.

The initialization coherence requirement of Definition 31 is guaranteed in an analogous way, by comparing initial cell codes and multi-cell codes.

Note that the first $n-1$ occurrences of alternations between universal and existential modalities $[\overline{E}]$ and $\langle\overline{\mathsf{E}}\rangle$ correspond to the alternations of universal and existential quantifications in the definition of alternating multi-tiling problem.

The correctness of the construction of $\varphi_{\mathcal{I}}$ is stated by the next proposition.

Proposition 32. *One can build, in time polynomial in the size of $\mathcal{I}$, $n+1$ $\mathsf{B}\overline{\mathsf{E}}$ formulas $\varphi_{IMT}, \varphi_1, \ldots, \varphi_n$ with $\Upsilon(\varphi_{IMT}) = \Upsilon(\varphi_1) = \ldots = \Upsilon(\varphi_n) = 0$, fulfilling the following conditions:*

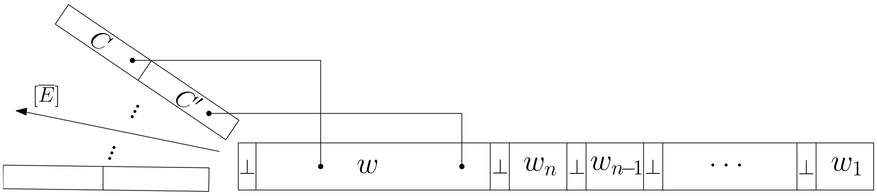


Fig. 7. Checking constraints between pairs of multi-cell codes C and C' in an initialized multi-tiling code.

- *for all finite words ρ over $\mathcal{AP}$ having the form $\rho = \rho' \cdot \perp \cdot w_n \cdots \perp \cdot w_1 \cdot end$ such that $\rho' \neq \varepsilon$ and $\perp \cdot w_n \cdots \perp \cdot w_1 \cdot end$ is a multi-initialization code, it holds $\mathcal{K}_\mathcal{I}, \rho \models \varphi_{IMT}$ if and only if ρ is an initialized multi-tiling code;*
- *for all $\ell \in [1, n]$ and words ρ having the form $\rho = \rho' \cdot \perp \cdot w_{\ell-1} \cdots \perp \cdot w_1 \cdot end$ such that $\rho' \neq \varepsilon$ and $w_j \in (\mathcal{AP} \setminus \{\perp\})^*$ for all $j \in [1, \ell - 1]$, it holds $\mathcal{K}_\mathcal{I}, \rho \models \varphi_\ell$ if and only if ρ' has the form $\rho' = \perp \cdot w_\ell$, where w_ℓ is an initialization code.*

Proof. Since each state of the Kripke structure $\mathcal{K}_\mathcal{I}$ is labeled by exactly one proposition letter of $\mathcal{AP}$, in the proof we exploit the standard regular expressions, where atomic formulas are single letters of $\mathcal{AP}$. Evidently, a standard regular expression can be converted into a propositional-based regular expression where each proposition letter $p \in \mathcal{AP}$ is replaced with the formula $p \wedge \bigwedge_{p' \in \mathcal{AP} \setminus \{p\}} \neg p'$. Let us focus on the construction of the $\mathsf{B}\overline{\mathsf{E}}$ formula φ_{IMT} (as $\varphi_1, \ldots, \varphi_n$ are simpler). First, we define a $\mathsf{B}\overline{\mathsf{E}}$ formula φ_{MT} ensuring the following property:

- for all finite words ρ over $\mathcal{AP}$ having the form $\rho = \rho' \cdot \perp \cdot w_n \cdot \ldots \cdot \perp \cdot w_1 \cdot end$ such that $\rho' \neq \varepsilon$ and $\perp \cdot w_n \cdot \ldots \cdot \perp \cdot w_1 \cdot end$ is a multi-initialization code, it holds $\mathcal{K}_\mathcal{I}, \rho \models \varphi_{MT}$ if and only if $\rho' = \perp \cdot w$ for some multi-tiling code w.

In order to build φ_{MT}, we need some auxiliary formulas.

- A regular expression $r_{mc} := D^n \cdot (\{r\} \times \{0, 1\})^n \cdot (\{c\} \times \{0, 1\})^n$ capturing the multi-cell codes.
- A B formula ψ_{comp} requiring that for each word $C \cdot \perp \cdot C_1 \cdot \ldots \cdot C_N \cdot \perp$ such that $C, C_1, \ldots, C_N$ are multi-cell codes, there is $i \in [1, N]$ such that C and C_i have the same row number and column number.

$$\psi_{comp} := \langle \mathsf{B} \rangle \Big((r_{mc} \cdot \perp \cdot (r_{mc})^+) \wedge \bigwedge_{i \in [1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{n+i-1} \cdot (r, b) \cdot \mathcal{AP}^+ \cdot (r, b) \cdot \mathcal{AP}^{2n-i}) \wedge \\ \bigwedge_{i \in [1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{2n+i-1} \cdot (c, b) \cdot \mathcal{AP}^+ \cdot (c, b) \cdot \mathcal{AP}^{n-i}) \Big)$$

- A propositional formula $\psi_=$ requiring that for each word having as a proper prefix $C \cdot C'$ such that C and C' are multi-cell codes, C and C' have the same row number and column number.

$$\psi_= := \bigwedge_{i \in [1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{n+i-1} \cdot (r, b) \cdot \mathcal{AP}^{3n-1} \cdot (r, b) \cdot \mathcal{AP}^+) \wedge \\ \bigwedge_{i \in [1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{2n+i-1} \cdot (c, b) \cdot \mathcal{AP}^{3n-1} \cdot (c, b) \cdot \mathcal{AP}^+)$$

- A propositional formula $\psi_{r,inc}$ (respectively, $\psi_{c,inc}$) requiring that for each word having as a proper prefix $C \cdot C'$ such that C and C' are multi-cell codes, C and C' have the same column number (respectively, the same row number), and there is $h \in [0, 2^n - 2]$ such that C and C' have row numbers (respectively, column numbers) h and $h + 1$, respectively. We consider the formula $\psi_{r,inc}$ (the definition of $\psi_{c,inc}$ is similar).

$$\psi_{r,inc} := \bigwedge_{i \in [1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{2n+i-1} \cdot (c, b) \cdot \mathcal{AP}^{3n-1} \cdot (c, b) \cdot \mathcal{AP}^+) \wedge \\ \bigvee_{i \in [1,n]} \Big(\bigwedge_{j \in [1,i-1]} (\mathcal{AP}^{n+j-1} \cdot (r, 1) \cdot \mathcal{AP}^{3n-1} \cdot (r, 0) \cdot \mathcal{AP}^+) \wedge \\ (\mathcal{AP}^{n+i-1} \cdot (r, 0) \cdot \mathcal{AP}^{3n-1} \cdot (r, 1) \cdot \mathcal{AP}^+) \wedge \\ \bigwedge_{j \in [i+1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{n+j-1} \cdot (r, b) \cdot \mathcal{AP}^{3n-1} \cdot (r, b) \cdot \mathcal{AP}^+) \Big)$$

- A B formula ψ_{double} requiring that for each word $C \cdot C' \cdot \perp \cdot C_1 \cdot \ldots \cdot C_N \cdot \perp$ such that $C, C', C_1, \ldots, C_N$ are multi-cell codes, there are $i, j \in [1, N]$ such that $C = C_i$ and $C' = C_j$; $\psi_{double} := \theta \wedge \theta'$, where θ (respectively, θ') requires that there is $i \in [1, N]$ such that $C_i = C$ (respectively, $C_i = C'$). We consider θ' (the definition of θ is similar).

$$\begin{aligned}\theta' := \langle \mathsf{B}\rangle\Big(&(r_{mc}\cdot r_{mc}\cdot\bot\cdot(r_{mc})^+) \;\wedge \bigwedge_{i\in[1,n]}\bigvee_{d\in D}(\mathcal{AP}^{3n+i-1}\cdot d\cdot\mathcal{AP}^+\cdot d\cdot\mathcal{AP}^{3n-i}) \;\wedge\\ &\bigwedge_{i\in[1,n]}\bigvee_{b\in\{0,1\}}(\mathcal{AP}^{4n+i-1}\cdot(r,b)\cdot\mathcal{AP}^+\cdot(r,b)\cdot\mathcal{AP}^{2n-i}) \;\wedge\\ &\bigwedge_{i\in[1,n]}\bigvee_{b\in\{0,1\}}(\mathcal{AP}^{5n+i-1}\cdot(c,b)\cdot\mathcal{AP}^+\cdot(c,b)\cdot\mathcal{AP}^{n-i})\Big)\end{aligned}$$

- A B formula ψ_{not_unique} requiring that for each word $C\cdot C'\cdot\bot\cdot C_1\cdot\ldots\cdot C_N\cdot\bot$ such that $C, C', C_1,\ldots,C_N$ are multi-cell codes, the following properties hold:
  - C and C' have the same row number and column number, but different content;
  - there are $i, j\in[1,N]$ such that $C=C_i$ and $C'=C_j$.

  The construction of ψ_{not_unique} is based on the formulas ψ_{double} and $\psi_=$:

$$\psi_{not\_unique} := \psi_{double}\wedge\psi_=\wedge\bigvee_{i\in[1,n]}\;\bigvee_{d,d'\in D:d\neq d'}(\mathcal{AP}^{i-1}\cdot d\cdot\mathcal{AP}^{3n-1}\cdot d'\cdot\mathcal{AP}^+).$$

- A B formula ψ_{row} (respectively, ψ_{col}) requiring that for each word $C\cdot C'\cdot\bot\cdot C_1\cdot\ldots\cdot C_N\cdot\bot$ such that $C, C', C_1,\ldots,C_N$ are multi-cell codes, the following condition holds.
  - Let us denote by $d_1\cdots d_n$ the content of C and by $d'_1\cdots d'_n$ the content of C'. Whenever (1) there are $i, j\in[1,N]$ such that $C=C_i$ and $C'=C_j$, and (2) C and C' have the same row number and column numbers h and $h+1$, respectively (respectively, C and C' have the same column number and row numbers h and $h+1$, respectively) for some $h\in[0,2^n-2]$, then it holds that $(d_\ell,d'_\ell)\in H$ (respectively, $(d_\ell,d'_\ell)\in V$), for all $\ell\in[1,N]$.

  We focus on ψ_{row} (the definition of ψ_{col} is similar):

$$\psi_{row} := (\psi_{double}\wedge\psi_{c,inc})\longrightarrow\bigwedge_{i\in[1,n]}\;\bigvee_{(d,d')\in H}(\mathcal{AP}^{i-1}\cdot d\cdot\mathcal{AP}^{3n-1}\cdot d'\cdot\mathcal{AP}^+).$$

Finally, the $\mathsf{B}\overline{\mathsf{E}}$ formula φ_{MT} is defined as follows:

$$\begin{aligned}&\neg(\mathcal{AP}^*\cdot\bot\cdot\mathcal{AP}^*)^{n+2}\;\wedge\;\langle \mathsf{B}\rangle\Big(\underbrace{(\bot\cdot(r_{mc})^+\cdot\bot)\wedge\neg\bigvee_{(d,d')\in D^2\setminus M}(\mathcal{AP}^+\cdot d\cdot d'\cdot\mathcal{AP}^+)}_{\text{Concatenation of well-formed multi-cell codes}}\;\wedge\\ &\underbrace{[\overline{E}]((r_{mc}\cdot\bot\cdot(r_{mc})^+\cdot\bot)\longrightarrow\psi_{comp})}_{\text{Completeness requirement of Definition 29}}\;\wedge\;\underbrace{[\overline{E}]((r_{mc}\cdot r_{mc}\cdot\bot\cdot(r_{mc})^+\cdot\bot)\longrightarrow\neg\psi_{not\_unique})}_{\text{Uniqueness requirement of Definition 29}}\;\wedge\\ &\underbrace{[\overline{E}]((r_{mc}\cdot r_{mc}\cdot\bot\cdot(r_{mc})^+\cdot\bot)\longrightarrow(\psi_{row}\wedge\psi_{col}))}_{\text{Row-adjacency and column-adjacency requirements of Definition 29}}\;\wedge\;\underbrace{\bigvee_{d_{acc}\in D_{acc}}(\mathcal{AP}^+\cdot d_{acc}\cdot(r,1)^n\cdot\mathcal{AP}^+)}_{\text{Acceptance requirement of Definition 29}}\Big).\end{aligned}$$

The $\mathsf{B}\overline{\mathsf{E}}$ formula φ_{IMT} is given by $\varphi_{MT}\wedge\varphi_{coh}$, where φ_{coh} ensures the initialization coherence requirement of Definition 31. In order to define φ_{coh}, we need some auxiliary formulas.

- A regular expression $r_{ic} := D_0\cdot(\{c\}\times\{0,1\})^n$ capturing the initial cell codes.
- A B formula ψ_{single} requiring that for each word $C\cdot\bot\cdot C_1\cdot\ldots\cdot C_N\cdot\bot$ such that $C, C_1,\ldots,C_N$ are multi-cell codes, there is $i\in[1,N]$ such that $C=C_i$ and the row number of C is 0. The definition of ψ_{single} is similar to the definition of ψ_{double}.
- A B formula ψ_{coh} requiring that for each word $C\cdot\bot\cdot C_1\cdot\ldots\cdot C_N\cdot\bot\cdot w_n\cdot\ldots\cdot\bot\cdot w_1\cdot end$ such that $C, C_1,\ldots,C_N$ are multi-cell codes and $\bot\cdot w_n\cdot\ldots\cdot\bot\cdot w_1\cdot end$ is a multi-initialization code, the following constraint holds: if there is $i\in[1,N]$ such that $C=C_i$, the row number of C is 0 and the content of C is $d_1\ldots d_n$, then for all $\ell\in[1,n]$, there exists an initial code in w_ℓ having the same column number as C and content d_ℓ.

$$\psi_{coh} := \big(\langle \mathsf{B}\rangle([(\mathcal{AP}\setminus\{\bot\})^+\cdot\bot\cdot(\mathcal{AP}\setminus\{\bot\})^+\cdot\bot]\wedge\psi_{single})\big)\longrightarrow\bigwedge_{\ell\in[1,n]}\psi_\ell;$$

$$\begin{aligned}\psi_\ell := \langle \mathsf{B}\rangle\Big(&\big[(\mathcal{AP}\setminus\{\bot\})^+\cdot(\bot\cdot(\mathcal{AP}\setminus\{\bot\})^+)^{n-\ell+1}\cdot\bot\cdot r_{ic}^+\big]\wedge\\ &\bigwedge_{i\in[1,n]}\bigvee_{b\in\{0,1\}}(\mathcal{AP}^{2n+i-1}\cdot(c,b)\cdot\mathcal{AP}^+\cdot(c,b)\cdot\mathcal{AP}^{n-i})\wedge\bigvee_{d\in D}(\mathcal{AP}^{\ell-1}\cdot d\cdot\mathcal{AP}^+\cdot d\cdot\mathcal{AP}^n)\Big)\end{aligned}$$

The $\mathsf{B\overline{E}}$ formula φ_{coh} is then $\varphi_{coh} := [\overline{E}]\big([r_{mc} \cdot (\bot \cdot (\mathcal{AP} \setminus \{\bot\})^+)^{n+1}] \to \psi_{coh}\big)$.

This concludes the proof of Proposition 32. □

Now recall that $\varphi_{\mathcal{I}} := [\overline{E}](\varphi_1 \to \langle \mathrm{\overline{E}} \rangle (\varphi_2 \wedge (\ldots ([\overline{E}](\varphi_{n-1} \to \langle \mathrm{\overline{E}} \rangle (\varphi_n \wedge \langle \mathrm{\overline{E}} \rangle\, \varphi_{IMT}))) \ldots)))$, where $\varphi_{IMT}, \varphi_1, \ldots, \varphi_n$ are the formulas defined in Proposition 32. Since the initial state of $\mathcal{K}_{\mathcal{I}}$ has no successors and the only initial trace has length 1 and corresponds to the proposition letter *end*, by Definitions 29–31, we have that $\mathcal{K}_{\mathcal{I}} \models \varphi_{\mathcal{I}}$ if and only if $\mathcal{I}$ is a positive instance of the alternating multi-tiling problem, proving Theorem 28. Now, this result combined with Theorem 24, implies the following complexity result.

Corollary 33. *The MC problem for* $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ *(and* $\mathsf{A\overline{A}E\overline{B}\overline{E}}$*) formulas over finite Kripke structures is* $\mathbf{AEXP_{pol}}$*-complete.*

6. The fragments $\mathsf{A\overline{A}B\overline{B}}$ and $\mathsf{A\overline{A}E\overline{E}}$

In this section we show that the two symmetric fragments $\mathsf{A\overline{A}B\overline{B}}$ and $\mathsf{A\overline{A}E\overline{E}}$ feature a better complexity, proving MC for them to be in **PSPACE**. To this end we first prove, in Section 6.1, that they enjoy an *exponential small-model property*, that is, if a trace ρ of a finite Kripke structure $\mathcal{K}$ satisfies a formula φ of $\mathsf{A\overline{A}B\overline{B}}/\mathsf{A\overline{A}E\overline{E}}$, then there is always a trace π, whose length is exponential in $|\varphi|$ and $|\mathcal{K}|$, that still satisfies φ. Therefore, without loss of generality, one can limit the verification of traces of $\mathcal{K}$ to those having at most exponential length. It is worth recalling that, in [6], we proved a *polynomial small-model property* in the sizes of the $\mathsf{A\overline{A}B\overline{B}}/\mathsf{A\overline{A}E\overline{E}}$ formula φ and the Kripke structure $\mathcal{K}$ *under the homogeneity assumption*.

Then, in Sections 6.2 and 6.3 we provide a **PSPACE** MC algorithm which exploits the exponential small-model property. Such an algorithm is completely different from the one presented in [6] for the MC problem of the same fragments, under the homogeneity assumption, which can exploit the aforementioned polynomial small-model property. As a matter of fact, unlike that of [6], this algorithm cannot store even a single—possibly exponential-length—trace, being bound to use polynomial working space. For this reason it visits the (exponential-length) traces of the input Kripke structure $\mathcal{K}$ by means of a binary reachability technique that allows it to use logarithmic space in the length of traces, hence guaranteeing the **PSPACE** complexity upper bound. The surprising fact is that both the algorithm of [6] and the one presented here use polynomial working space, thus showing that relaxing the homogeneity assumption comes at no additional computational cost for the fragments $\mathsf{A\overline{A}B\overline{B}}$ and $\mathsf{A\overline{A}E\overline{E}}$.

Finally, in Section 6.3 we prove the **PSPACE**-completeness of MC for $\mathsf{A\overline{A}B\overline{B}}$ and $\mathsf{A\overline{A}E\overline{E}}$.

6.1. Exponential small-model property for $\mathsf{A\overline{A}B\overline{B}}$ *and* $\mathsf{A\overline{A}E\overline{E}}$

In this section we prove the exponential small-model property for the fragments $\mathsf{A\overline{A}B\overline{B}}$ and $\mathsf{A\overline{A}E\overline{E}}$ (actually, we focus only on $\mathsf{A\overline{A}B\overline{B}}$ being the case for $\mathsf{A\overline{A}E\overline{E}}$ symmetric).

Given a DFA $\mathcal{D} = (\Sigma, Q, q_0, \delta, F)$, we denote by $\mathcal{D}(w)$ (resp., $\mathcal{D}_q(w)$) the state reached by the computation of $\mathcal{D}$ from q_0 (resp., $q \in Q$) over the word $w \in \Sigma^*$. We now consider *well-formedness* of induced traces (recall Definition 22) w.r.t. a set of DFAs: a well formed trace π induced by ρ preserves the states of the computations of the DFAs reached by reading prefixes of ρ and π bounded by corresponding positions.

Definition 34 *(Well-formed trace).* Let $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ be a finite Kripke structure, $\rho \in \mathrm{Trc}_{\mathcal{K}}$ be a trace, and $\mathcal{D}^s = (2^{\mathcal{AP}}, Q^s, q_0^s, \delta^s, F^s)$ with $s = 1, \ldots, k$, be DFAs. A trace $\pi \in \mathrm{Trc}_{\mathcal{K}}$ *induced by* ρ is $(q^1_{\ell_1}, \ldots, q^k_{\ell_k})$*-well-formed w.r.t.* ρ, with $q^s_{\ell_s} \in Q^s$ for all $s = 1, \ldots, k$, if and only if:

- for all π-positions j, with corresponding ρ-positions i_j, and all $s = 1, \ldots, k$, it holds that $\mathcal{D}^s_{q^s_{\ell_s}}(\mu(\pi^j)) = \mathcal{D}^s_{q^s_{\ell_s}}(\mu(\rho^{i_j}))$.

It is easy to see that, for $q^s_{\ell_s} \in Q^s$, $s = 1, \ldots, k$, the $(q^1_{\ell_1}, \ldots, q^k_{\ell_k})$-well-formedness relation is *transitive*.

Now it is possible to show that a trace whose length exceeds a suitable exponential threshold, induces a shorter, well-formed trace. Such a contraction pattern (Proposition 35) represents a "basic step" in a contraction process which will allow us to prove the exponential small-model property for $\mathsf{A\overline{A}B\overline{B}}$.

Let us consider an $\mathsf{A\overline{A}B\overline{B}}$ formula φ and let $r_1, \ldots, r_k$ be the RE's over $\mathcal{AP}$ in φ. Let $\mathcal{D}^1, \ldots, \mathcal{D}^k$ be the DFAs such that $\mathcal{L}(\mathcal{D}^t) = \mathcal{L}(r_t)$, for $t = 1, \ldots, k$, where $|Q^t| \leq 2^{2|r_t|}$ (see Remark 3). We denote $Q^1 \times \ldots \times Q^k$ by $Q(\varphi)$, and $\mathcal{D}^1, \ldots, \mathcal{D}^k$ by $\mathcal{D}(\varphi)$.

Proposition 35. *Let* $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ *be a finite Kripke structure,* φ *be an* $\mathsf{A\overline{A}B\overline{B}}$ *formula with* RE*'s* $r_1, \ldots, r_k$ *over* $\mathcal{AP}$*,* $\rho \in \mathrm{Trc}_{\mathcal{K}}$ *be a trace, and* $(q^1, \ldots, q^k) \in Q(\varphi)$*. There exists a trace* $\pi \in \mathrm{Trc}_{\mathcal{K}}$*, which is* $(q^1, \ldots, q^k)$*-well-formed w.r.t.* ρ*, such that* $|\pi| \leq |S| \cdot 2^{2\sum_{\ell=1}^{k} |r_\ell|}$.

Proof. Let $\rho \in \mathrm{Trc}_{\mathcal{K}}$ with $|\rho| = n$. If $n \leq |S| \cdot 2^{2\sum_{\ell=1}^{k} |r_\ell|}$, the thesis trivially holds. Thus, let us assume $n > |S| \cdot 2^{2\sum_{\ell=1}^{k} |r_\ell|}$. We show that there exists a trace which is $(q^1, \ldots, q^k)$-well-formed w.r.t. ρ, whose length is smaller than n. The number of

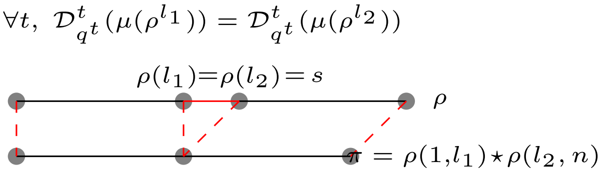

$\forall t,\ \mathcal{D}^t_{q^t}(\mu(\rho^{l_1})) = \mathcal{D}^t_{q^t}(\mu(\rho^{l_2}))$
$\rho(l_1){=}\rho(l_2){=}s$
ρ
$\pi = \rho(1,l_1)\star\rho(l_2,n)$

Fig. 8. The contraction step of Proposition 35.

possible (joint) configurations of the DFAs $\mathcal{D}(\varphi)$ is (at most) $|Q(\varphi)| \le 2^{2^{|r_1|}} \cdots 2^{2^{|r_k|}} = 2^{2^{\sum_{\ell=1}^{k}|r_\ell|}}$. Since $n > |S| \cdot 2^{2^{\sum_{\ell=1}^{k}|r_\ell|}}$, there exists some state $s \in S$ occurring in ρ at least twice in the ρ-positions say $1 \le l_1 < l_2 \le |\rho|$, such that $\mathcal{D}^t_{q^t}(\mu(\rho^{l_1})) = \mathcal{D}^t_{q^t}(\mu(\rho^{l_2}))$, for all $t = 1, \dots, k$. Let us consider $\pi = \rho(1, l_1) \star \rho(l_2, n)$ (see Fig. 8). It is easy to see that $\pi \in \mathrm{Trc}_{\mathcal{K}}$, as $\rho(l_1) = \rho(l_2)$, and $|\pi| < n$. Moreover, π is $(q^1, \dots, q^k)$-well-formed w.r.t. ρ (the corresponding positions are $i_j = j$ if $j \le l_1$, and $i_j = j + (l_2 - l_1)$ otherwise). Now, if $|\pi| \le |S| \cdot 2^{2^{\sum_{\ell=1}^{k}|r_\ell|}}$, the thesis holds. Otherwise, the same basic step can be iterated a finite number of times: the thesis follows by transitivity of the $(q^1, \dots, q^k)$-well-formedness relation. □

The next step is to determine some conditions for contracting traces while preserving the equivalence w.r.t. the satisfiability of the considered $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$ formula. In the following, we restrict ourselves to formulas in NNF. For a trace ρ and a formula φ of $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$ (in NNF), we fix some special ρ-positions, called *witness positions*, each one corresponding to the minimal prefix of ρ which satisfies a formula ψ occurring in φ as a subformula of the form $\langle \mathrm{B} \rangle \psi$ (provided that $\langle \mathrm{B} \rangle \psi$ is satisfied by ρ). As we will see in the proof of Theorem 37, when a contraction is performed in between a pair of *consecutive* witness positions (thus no witness position is ever removed), we get a trace induced by ρ (according to Definition 22) equivalent w.r.t. the satisfiability of φ.

Definition 36 *(Witness positions).* Let ρ be a trace of $\mathcal{K}$ and φ be a formula of $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$. Let us denote by $B(\varphi, \rho)$ the set of subformulas $\langle \mathrm{B} \rangle \psi$ of φ such that $\mathcal{K}, \rho \models \langle \mathrm{B} \rangle \psi$. The *set* $Wt(\varphi, \rho)$ *of witness positions of* ρ *for* φ is the *minimal* set of ρ-positions satisfying the following constraint: for each $\langle \mathrm{B} \rangle \psi \in B(\varphi, \rho)$, the smallest ρ-position $i < |\rho|$ such that $\mathcal{K}, \rho^i \models \psi$ belongs to $Wt(\varphi, \rho)$.[5]

Clearly, the cardinality of $B(\varphi, \rho)$ and of $Wt(\varphi, \rho)$ is at most $|\varphi| - 1$. We are finally ready to prove the exponential small-model property for $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$.

Theorem 37 *(Exponential small-model for $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$). Let $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$, $\sigma, \rho \in \mathrm{Trc}_{\mathcal{K}}$, and φ be an $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$ formula in NNF, with RE's $r_1, \dots, r_u$ over $\mathcal{AP}$, such that $\mathcal{K}, \sigma \star \rho \models \varphi$. Then, there exists $\pi \in \mathrm{Trc}_{\mathcal{K}}$, induced by ρ, such that $\mathcal{K}, \sigma \star \pi \models \varphi$ and $|\pi| \le |S| \cdot (|\varphi| + 1) \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$.*

The theorem holds in particular if $|\sigma| = 1$, and thus $\sigma \star \rho = \rho$ and $\sigma \star \pi = \pi$. In this case, if $\mathcal{K}, \rho \models \varphi$, then $\mathcal{K}, \pi \models \varphi$, where π is induced by ρ and $|\pi| \le |S| \cdot (|\varphi| + 1) \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$. The more general assertion is needed for technical reasons (in the soundness/completeness proof of the next algorithms, see Theorem 40).

Proof. Let $Wt(\varphi, \sigma \star \rho)$ be the set of witness positions of $\sigma \star \rho$ for φ. Let $\{i_1, \dots, i_k\}$ be the ordering of $Wt(\varphi, \sigma \star \rho)$ such that $i_1 < \ldots < i_k$. Let $i_0 = 1$ and $i_{k+1} = |\sigma \star \rho|$. Hence, $1 = i_0 \le i_1 < \ldots < i_k < i_{k+1} = |\sigma \star \rho|$. If the length of ρ is at most $|S| \cdot (|\varphi| + 1) \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$, the thesis trivially holds. Let us assume that $|\rho| > |S| \cdot (|\varphi| + 1) \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$. We show that there exists a trace π induced by ρ, with $|\pi| < |\rho|$, such that $\mathcal{K}, \sigma \star \pi \models \varphi$.

W.l.o.g., we can assume that $i_0 \le i_1 < \ldots < i_{j-1}$, for some $j \ge 1$, are σ-positions (while $i_j < \ldots < i_{k+1}$ are $(\sigma \star \rho)$-positions not in σ). We claim that either (i) there exists $t \in [j, k]$ such that $i_{t+1} - i_t > |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$ or (ii) $|(\sigma \star \rho)(|\sigma|, i_j)| > |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$. By way of contradiction, suppose that neither (i) nor (ii) holds. We need to distinguish two cases. If $\sigma \star \rho = \rho$, then $|\rho| = (i_{k+1} - i_0) + 1 \le (k+1) \cdot |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}} + 1$ (a contradiction); otherwise ($|\rho| < |\sigma \star \rho|$), $|\rho| = (i_{k+1} - i_j) + |(\sigma \star \rho)(|\sigma|, i_j)| \le k \cdot |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}} + |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}} \le (k+1) \cdot |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$. The contradiction follows since $(k+1) \cdot |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}} + 1 \le |\varphi| \cdot |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}} + 1 \le |S| \cdot (|\varphi| + 1) \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$.

Let us define $(\alpha, \beta) = (i_t, i_{t+1})$ in case (i), and $(\alpha, \beta) = (|\sigma|, i_j)$ in case (ii). Moreover let $\rho' = \rho(\alpha, \beta)$. In both the cases, we have $|\rho'| > |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}}$. By Proposition 35, there exists a trace π' of $\mathcal{K}$, $(q^1, \cdots, q^u)$-well-formed with respect to ρ', such that $|\pi'| \le |S| \cdot 2^{2^{\sum_{\ell=1}^{u}|r_\ell|}} < |\rho'|$, where we choose $q^x = \mathcal{D}^x(\mu((\sigma \star \rho)^{\alpha-1}))$ for $x = 1, \dots, u$ (as a particular case we set q^x as the initial state of $\mathcal{D}^x$ if $\alpha = 1$). Let π be the trace induced by ρ obtained by replacing the subtrace ρ' of ρ with π' (see Fig. 9). Since $|\pi| < |\rho|$, it remains to prove that $\mathcal{K}, \sigma \star \pi \models \varphi$.

[5] Note that such a ρ-position exists by definition of $B(\varphi, \rho)$.

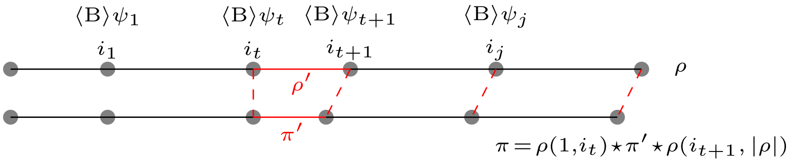


Fig. 9. Representation of the contraction step of Theorem 37—case (*i*).

Let us denote $\sigma\star\pi$ by $\overline{\pi}$ and $\sigma\star\rho$ by $\overline{\rho}$. Moreover, let $H:[1,|\overline{\pi}|]\to[1,|\overline{\rho}|]$ be the function mapping positions of $\overline{\pi}$ into positions of $\overline{\rho}$ in this way: positions "outside" π' (i.e., outside the interval $[\alpha,\alpha+|\pi'|-1]$) are mapped into their original position in $\overline{\rho}$; positions "inside" π' (i.e., in $[\alpha,\alpha+|\pi'|-1]$) are mapped to the corresponding position in ρ' (exploiting well-formedness of π' w.r.t. to ρ'). Formally, H is defined as:

$$H(m)=\begin{cases} m & \text{if} \quad m<\alpha \\ \alpha+\ell_{m-\alpha+1}-1 & \text{if} \quad \alpha\le m<\alpha+|\pi'| \\ m+(|\rho'|-|\pi'|) & \text{if} \quad m\ge\alpha+|\pi'| \end{cases} \tag{1}$$

where ℓ_m is the ρ'-position corresponding to the π'-position m. It is easy to check that H satisfies the following properties:

1. H is strictly monotonic, i.e., for all $j,j'\in[1,|\overline{\pi}|]$, $j<j'$ iff $H(j)<H(j')$;
2. for all $j\in[1,|\overline{\pi}|]$, $\overline{\pi}(j)=\overline{\rho}(H(j))$;
3. $H(1)=1$ and $H(|\overline{\pi}|)=|\overline{\rho}|$;
4. $Wt(\varphi,\overline{\rho})\subseteq\{H(j)\mid j\in[1,|\overline{\pi}|]\}$;
5. for each $j\in[1,|\overline{\pi}|]$ and $x=1,\dots,u$, $\mathcal{D}^x(\mu(\overline{\pi}^j))=\mathcal{D}^x(\mu(\overline{\rho}^{H(j)}))$.

We only comment on Property 5. The property holds for $j\in[1,\alpha-1]$, as $\overline{\pi}^j=\overline{\rho}^{H(j)}=\overline{\rho}^j$. For $j\in[\alpha,\alpha+|\pi'|-1]$, $\mathcal{D}^x(\mu(\overline{\pi}^j))=\mathcal{D}^x(\mu(\overline{\rho}^{H(j)}))$ follows from the well-formedness hypothesis. Finally, being $\overline{\rho}(\beta,|\overline{\rho}|)=\overline{\pi}(\alpha+|\pi'|-1,|\overline{\pi}|)$ and $\mathcal{D}^x(\mu(\overline{\pi}^{\alpha+|\pi'|-1}))=\mathcal{D}^x(\mu(\overline{\rho}^{\beta}))$, the property holds also for $j\in[\alpha+|\pi'|,|\overline{\pi}|]$.

The statement $\mathcal{K},\overline{\pi}\models\varphi$ is an immediate consequence of the following claim, considering that $H(|\overline{\pi}|)=|\overline{\rho}|$, $\mathcal{K},\overline{\rho}\models\varphi$, $\overline{\rho}^{|\overline{\rho}|}=\overline{\rho}$, and $\overline{\pi}^{|\overline{\pi}|}=\overline{\pi}$.

Claim 38. For all $j\in[1,|\overline{\pi}|]$, all subformulas ψ of φ, and all $\xi\in\mathrm{Trc}_{\mathcal{K}}$,

$$\mathcal{K},\overline{\rho}^{H(j)}\star\xi\models\psi\implies\mathcal{K},\overline{\pi}^{j}\star\xi\models\psi.$$

Proof. Assume that $\mathcal{K},\overline{\rho}^{H(j)}\star\xi\models\psi$. Note that $\overline{\rho}^{H(j)}\star\xi$ is defined if and only if $\overline{\pi}^j\star\xi$ is defined. We prove by induction on the structure of ψ that $\mathcal{K},\overline{\pi}^j\star\xi\models\psi$. Since φ is in NNF, only the following cases can occur.

(Base) case $\psi=r_t$ or $\psi=\neg r_t$ where r_t is some RE over $\mathcal{AP}$. By Property 5 of H, $\mathcal{D}^t(\mu(\overline{\pi}^j))=\mathcal{D}^t(\mu(\overline{\rho}^{H(j)}))$, thus $\mathcal{D}^t(\mu(\overline{\pi}^j\star\xi))=\mathcal{D}^t(\mu(\overline{\rho}^{H(j)}\star\xi))$. It follows that $\mathcal{K},\overline{\pi}^j\star\xi\models r_t$ if and only if $\mathcal{K},\overline{\rho}^{H(j)}\star\xi\models r_t$, and the result holds.

Case $\psi=\theta_1\wedge\theta_2$ or $\psi=\theta_1\vee\theta_2$. The result holds by the inductive hypothesis.

Case $\psi=[B]\theta$. We need to show that for each proper prefix η of $\overline{\pi}^j\star\xi$, we have $\mathcal{K},\eta\models\theta$. We distinguish two cases:

- η is *not* a proper prefix of $\overline{\pi}^j$. Hence, η has the form $\overline{\pi}^j\star\xi^h$ for some $h\in[1,|\xi|-1]$. Since $\mathcal{K},\overline{\rho}^{H(j)}\star\xi\models[B]\theta$, then $\mathcal{K},\overline{\rho}^{H(j)}\star\xi^h\models\theta$. By the inductive hypothesis, we have $\mathcal{K},\overline{\pi}^j\star\xi^h\models\theta$.
- η is a proper prefix of $\overline{\pi}^j$. Hence, $\eta=\overline{\pi}^h$ for some $h\in[1,j-1]$. By Property 1 of H, $H(h)<H(j)$, and since $\mathcal{K},\overline{\rho}^{H(j)}\star\xi\models[B]\theta$, we have that $\mathcal{K},\overline{\rho}^{H(h)}\models\theta$. By the inductive hypothesis, $\mathcal{K},\overline{\pi}^h\models\theta$.

Case $\psi=\langle\mathrm{B}\rangle\,\theta$. We have to show that there exists a proper prefix of $\overline{\pi}^j\star\xi$ satisfying θ. Since $\mathcal{K},\overline{\rho}^{H(j)}\star\xi\models\psi$, there exists a proper prefix η' of $\overline{\rho}^{H(j)}\star\xi$ such that $\mathcal{K},\eta'\models\theta$. We distinguish two cases:

- η' is *not* a proper prefix of $\overline{\rho}^{H(j)}$. Hence, η' is of the form $\overline{\rho}^{H(j)}\star\xi^h$ for some $h\in[1,|\xi|-1]$. By the inductive hypothesis, $\mathcal{K},\overline{\pi}^j\star\xi^h\models\theta$, and $\mathcal{K},\overline{\pi}^j\star\xi\models\langle\mathrm{B}\rangle\,\theta$.
- η' is a proper prefix of $\overline{\rho}^{H(j)}$. Hence, $\eta'=\overline{\rho}^i$ for some $i\in[1,H(j)-1]$, and $\mathcal{K},\overline{\rho}^i\models\theta$. Let i' be the smallest position of $\overline{\rho}$ such that $\mathcal{K},\overline{\rho}^{i'}\models\theta$. Hence $i'\le i$ and, by Definition 36, $i'\in Wt(\varphi,\overline{\rho})$. By Property 4 of H, $i'=H(h)$ for some $\overline{\pi}$-position h. Since $H(h)<H(j)$, it holds that $h<j$ (Property 1). By the inductive hypothesis, $\mathcal{K},\overline{\pi}^h\models\theta$, and $\mathcal{K},\overline{\pi}^j\star\xi\models\langle\mathrm{B}\rangle\,\theta$.

Cases $\psi=[\overline{B}]\theta$ or $\psi=\langle\overline{\mathrm{B}}\rangle\,\theta$. The thesis directly follows from the inductive hypothesis.

Cases $\psi=[A]\theta$, $\psi=\langle\mathrm{A}\rangle\,\theta$, $\psi=[\overline{A}]\theta$ or $\psi=\langle\overline{\mathrm{A}}\rangle\,\theta$. Since $\overline{\pi}^j\star\xi$ and $\overline{\rho}^{H(j)}\star\xi$ start at the same state and lead to the same state (by Property 2 and 3 of H), the result trivially follows, concluding the proof of the claim. □

We have shown that $\mathcal{K}, \overline{\pi} \models \varphi$, with $|\pi| < |\rho|$. If $|\pi| \leq |S| \cdot (|\varphi| + 1) \cdot 2^{2\sum_{\ell=1}^{u} |r_\ell|}$, the thesis holds. Otherwise, we can iterate the above contraction a finite number of times until the bound is reached. □

The proved exponential small-model property allows us to devise a trivial *exponential working space* algorithm for $\mathsf{A\overline{A}B\overline{B}}$ (and $\mathsf{A\overline{A}E\overline{E}}$) (actually we shall present a polynomial space one in the next sections), which basically unravels the Kripke structure and checks all the subformulas of the input formula. At every step it can consider traces not longer than $O(|S| \cdot |\varphi| \cdot 2^{2\sum_{\ell=1}^{u} |r_\ell|})$. Conversely, the following example shows that the exponential small-model is strict, that is, there exists a formula and a Kripke structure, such that the shortest trace satisfying the formula has exponential length in the size of the formula itself. This is the case even for pure propositional formulas.

Example 39. Let pr_i be the i-th smallest prime number. It is well-known that $pr_i \in O(i \log i)$. Let $w^{\otimes k}$ denote the string obtained by concatenating k times w. Let us fix some $n \in \mathbb{N}$, and let $\mathcal{K} = (\{p\}, \{s\}, R, \mu, s)$ be the trivial Kripke structure having only one state with a self-loop, where $R = \{(s, s)\}$, and $\mu(s) = \{p\}$. The shortest trace satisfying $\psi = \bigwedge_{i=1}^{n} (p^{\otimes(pr_i)})^*$ is $\rho = s^{\otimes(pr_1 \cdots pr_n)}$, since its length is the least common multiple of $pr_1, \ldots, pr_n$, which is indeed $pr_1 \cdots pr_n$. It is immediate to check that the length of ψ is $O(n \cdot pr_n) = O(n^2 \log n)$. On the other hand, the length of ρ is $pr_1 \cdots pr_n \geq 2^n$.

In the following, we will exploit the exponential small-model property of the two symmetrical fragments $\mathsf{A\overline{A}B\overline{B}}$ and $\mathsf{A\overline{A}E\overline{E}}$ to prove the **PSPACE**-completeness of their MC problems. First, in Section 6.2, we will provide a **PSPACE** MC algorithm for $\mathsf{B\overline{B}}$ (resp., $\mathsf{E\overline{E}}$). Then, in Section 6.3, we will show that the *meets* and *met-by* modalities A and $\mathsf{\overline{A}}$ can be suitably encoded by regular expressions without increasing the complexity of $\mathsf{B\overline{B}}$ (resp., $\mathsf{E\overline{E}}$).

6.2. **PSPACE***-membership of MC for* $\mathsf{B\overline{B}}$

In this section, to start with, we describe a **PSPACE** MC algorithm for $\mathsf{B\overline{B}}$ formulas. W.l.o.g., we assume that the processed formulas do not contain occurrences of the universal modalities $[B]$ and $[\overline{B}]$. Moreover, for a formula ψ, we denote by $\mathsf{Subf}_{\langle \mathrm{B} \rangle}(\psi) = \{\varphi \mid \langle \mathrm{B} \rangle\, \varphi \text{ is a subformula of } \psi\}$. In such an algorithm, Φ represents the overall formula to be checked, while the parametric formula ψ ranges over its subformulas.

Due to the result of the previous section, the algorithm can consider only traces having length bounded by the exponential small-model property. Note that an algorithm required to work in polynomial space cannot explicitly store the DFAs for the regular expressions occurring in Φ (their states are *exponentially* many in the length of the associated regular expressions). For this reason, while checking a formula against a trace, the algorithm just stores the *current states* of the computations of the DFAs associated with the regular expressions in Φ, from the respective initial states (in the following such states are denoted—with a little abuse of notation—again by $\mathcal{D}(\Phi)$, and called the *"current configuration" of the DFAs*) and calculates on-the-fly the successor states in the DFAs, once they have read some state of $\mathcal{K}$ used to extend the considered trace (this can be done by exploiting a *succinct* encoding of the NFAs for the regular expressions of Φ, see Remark 3 in Section 2).

A call to the recursive procedure `Check`$(\mathcal{K}, \psi, s, G, \mathcal{D}(\Phi))$ (Algorithm 3) checks the satisfiability of a subformula ψ of Φ w.r.t. any trace ρ fulfilling the following conditions:

1. $G \subseteq \mathsf{Subf}_{\langle \mathrm{B} \rangle}(\psi)$ is the set of formulas that hold true on at least a prefix of ρ;
2. after reading $\mu(\rho(1, |\rho| - 1))$ the current configuration of the DFAs for the regular expressions of Φ is $\mathcal{D}(\Phi)$;
3. the last state of ρ is s.

Intuitively, since the algorithm cannot store the already checked portion of a trace (whose length could be exponential), the relevant information is *summarized* in a triple $(G, \mathcal{D}(\Phi), s)$. Hereafter the set of all possible summarizing triples $(\overline{G}, \overline{\mathcal{D}(\Phi)}, \overline{s})$, where $\overline{G} \subseteq \mathsf{Subf}_{\langle \mathrm{B} \rangle}(\psi)$, $\overline{\mathcal{D}(\Phi)}$ is any current configuration of the DFAs for the regular expressions of Φ, and $\overline{s}$ is a state of $\mathcal{K}$, is denoted by $\mathsf{Conf}(\mathcal{K}, \psi)$.

Let us consider in detail the body of the procedure. First `advance`$(\mathcal{D}(\Phi), \mu(s))$, invoked at line 2, updates the current configuration of the DFAs after reading the symbol $\mu(s)$. If ψ is a regular expression r (lines 1–5), we just check whether the (computation of the) DFA associated with r is in a final state (i.e., the summarized trace is accepted). Boolean connectives are easily dealt with recursively (lines 6–9). If ψ has the form $\langle \mathrm{B} \rangle\, \psi'$ (lines 10–14), then ψ' has to hold over a proper prefix of the summarized trace, i.e. ψ' must belong to G.

The only involved case is $\psi = \langle \overline{\mathrm{B}} \rangle\, \psi'$ (lines 15–19): we have to unravel the Kripke structure $\mathcal{K}$ to find an *extension* ρ' of ρ, summarized by the triple $(G', \mathcal{D}(\Phi)', s')$, satisfying ψ'. The idea is checking whether or not there exists a summarized trace $(G', \mathcal{D}(\Phi)', s')$, suitably extending $(G, \mathcal{D}(\Phi), s)$, namely, such that:

1. $\mathcal{D}(\Phi)'$ and s' are *synchronously* reachable from $\mathcal{D}(\Phi)$ and s, respectively;
2. $G' \supseteq G$ contains any formula of $\mathsf{Subf}_{\langle \mathrm{B} \rangle}(\psi')$ satisfied by a prefix of the extension;
3. the extension $(G', \mathcal{D}(\Phi)', s')$ satisfies ψ'.

Algorithm 3 Check($\mathcal{K}, \psi, s, G, \mathcal{D}(\Phi)$)

```
1: if $\psi = r$ then                                        ▷ $r$ is a regular expression
2:   if the current state of the DFA for $r$ in advance($\mathcal{D}(\Phi), \mu(s)$) is final then
3:     return $\top$
4:   else
5:     return $\bot$
6: else if $\psi = \neg\psi'$ then
7:   return not Check($\mathcal{K}, \psi', s, G, \mathcal{D}(\Phi)$)
8: else if $\psi = \psi_1 \wedge \psi_2$ then
9:   return Check($\mathcal{K}, \psi_1, s, G \cap \mathsf{Subf}_{\langle B \rangle}(\psi_1), \mathcal{D}(\Phi)$) and Check($\mathcal{K}, \psi_2, s, G \cap \mathsf{Subf}_{\langle B \rangle}(\psi_2), \mathcal{D}(\Phi)$)
10: else if $\psi = \langle B \rangle\, \psi'$ then
11:   if $\psi' \in G$ then
12:     return $\top$
13:   else
14:     return $\bot$
15: else if $\psi = \langle \overline{B} \rangle\, \psi'$ then
16:   for each $b \in \{1, \dots, |S| \cdot (2|\psi'|+1) \cdot 2^{2^{\sum_{\ell=1}^{u} |r_\ell|}} - 1\}$ and each $(G', \mathcal{D}(\Phi)', s') \in \mathsf{Conf}(\mathcal{K}, \psi)$ do▷ $r_1, \dots, r_u$ are the regular expressions of $\psi'$
17:     if Reach($\mathcal{K}, \psi', (G, \mathcal{D}(\Phi), s), (G', \mathcal{D}(\Phi)', s'), b$) and Check($\mathcal{K}, \psi', s', G', \mathcal{D}(\Phi)'$) then
18:       return $\top$
19:   return $\bot$
```

In order to check point (1), i.e., synchronous reachability, we can exploit the exponential small-model property and consider only the unraveling of $\mathcal{K}$ starting from s having depth at most $|S| \cdot (2|\psi'|+1) \cdot 2^{2^{\sum_{\ell=1}^{u} |r_\ell|}} - 1$.[6] The check of (1) and (2) is performed by the procedure Reach (Algorithm 4), which accepts as input two summarized traces and a bound b on the depth of the unraveling of $\mathcal{K}$. The proposed reachability algorithm is reminiscent of the binary reachability of Savitch's theorem [17].

Algorithm 4 Reach($\mathcal{K}, \psi, (G_1, \mathcal{D}(\Phi)_1, s_1), (G_2, \mathcal{D}(\Phi)_2, s_2), b$)

```
1: if $b = 1$ then
2:   return Compatible($\mathcal{K}, \psi, (G_1, \mathcal{D}(\Phi)_1, s_1), (G_2, \mathcal{D}(\Phi)_2, s_2)$)
3: else                                                     ▷ $b \geq 2$
4:   $b' \leftarrow \lfloor b/2 \rfloor$
5:   for each $(G_3, \mathcal{D}(\Phi)_3, s_3) \in \mathsf{Conf}(\mathcal{K}, \psi)$ do
6:     if Reach($\mathcal{K}, \psi, (G_1, \mathcal{D}(\Phi)_1, s_1), (G_3, \mathcal{D}(\Phi)_3, s_3), b'$) and Reach($\mathcal{K}, \psi, (G_3, \mathcal{D}(\Phi)_3, s_3), (G_2, \mathcal{D}(\Phi)_2, s_2), b - b'$) then
7:       return $\top$
8:   return $\bot$
```

The procedure Reach proceeds recursively (lines 3–8) by halving at each step the value b of the length bound, until it gets called over two states s_1 and s_2 which are adjacent in a trace. At each halving step, an intermediate summarizing triple is generated to be associated with the split point. At the base of recursion (for $b = 1$, lines 1–2), the auxiliary procedure Compatible (Algorithm 5) is invoked. At line 1, Compatible checks whether there is an edge between s_1 and s_2 ($(s_1, s_2) \in R$), and if, at the considered step, the current configuration of the DFAs $\mathcal{D}(\Phi)_1$ is transformed into the configuration $\mathcal{D}(\Phi)_2$ (i.e., s_2 and $\mathcal{D}(\Phi)_2$ are synchronously reachable from s_1 and $\mathcal{D}(\Phi)_1$). At lines 2–9, Compatible checks that each formula φ in $(G_2 \setminus G_1)$, where $G_2 \supseteq G_1$, is satisfied by a trace summarized by $(G_1, \mathcal{D}(\Phi)_1, s_1)$ (lines 2–5). Intuitively, $(G_1, \mathcal{D}(\Phi)_1, s_1)$ summarizes the maximal prefix of $(G_2, \mathcal{D}(\Phi)_2, s_2)$, and thus a subformula satisfied by a prefix of a trace summarized by $(G_2, \mathcal{D}(\Phi)_2, s_2)$ either belongs to G_1 or it is satisfied by the trace summarized by $(G_1, \mathcal{D}(\Phi)_1, s_1)$. Moreover, (lines 6–9) Compatible checks that G_2 is maximal (i.e., no subformula that must be in G_2 has been forgotten). Note that by exploiting this binary reachability technique, the recursion depth of Reach is logarithmic in the length of the trace to be visited, hence it can use only polynomial space. Theorem 40 establishes the soundness of procedure Check.

Theorem 40. *Let Φ be a $\mathsf{B\overline{B}}$ formula, ψ be a subformula of Φ, and $\rho \in \mathrm{Trc}_{\mathcal{K}}$ be a trace with $s = \mathrm{lst}(\rho)$. Let G be the subset of formulas in $\mathsf{Subf}_{\langle B \rangle}(\psi)$ that hold on some proper prefix of ρ. Let $\mathcal{D}(\Phi)$ be the current configuration of the DFAs associated with the regular expressions in Φ after reading $\mu(\rho(1, |\rho| - 1))$. Then, Check($\mathcal{K}, \psi, s, G, \mathcal{D}(\Phi)$) $= \top \iff \mathcal{K}, \rho \models \psi$.*

Proof. The proof is by induction on the structure of ψ. The thesis trivially follows for the cases $\psi = r$ (regular expression), $\psi = \neg\psi'$, $\psi = \psi_1 \wedge \psi_2$, and $\psi = \langle B \rangle\, \psi'$.

[6] The factor 2 of $|\psi'|$ is added since the exponential small-model for $\mathsf{A\overline{A}B\overline{B}}$ requires a formula in NNF.

Algorithm 5 $\texttt{Compatible}(\mathcal{K}, \psi, (G_1, \mathcal{D}(\Phi)_1, s_1), (G_2, \mathcal{D}(\Phi)_2, s_2))$

```
if $(s_1, s_2) \in R$ and $\texttt{advance}(\mathcal{D}(\Phi)_1, \mu(s_1)) = \mathcal{D}(\Phi)_2$ and $G_1 \subseteq G_2$ then
   for each $\varphi \in (G_2 \setminus G_1)$ do
      $G \leftarrow G_1 \cap \mathsf{Subf}_{\langle B \rangle}(\varphi)$
      if $\texttt{Check}(\mathcal{K}, \varphi, s_1, G, \mathcal{D}(\Phi)_1) = \bot$ then
         return $\bot$
   for each $\varphi \in (\mathsf{Subf}_{\langle B \rangle}(\psi) \setminus G_2)$ do
      $G \leftarrow G_1 \cap \mathsf{Subf}_{\langle B \rangle}(\varphi)$
      if $\texttt{Check}(\mathcal{K}, \varphi, s_1, G, \mathcal{D}(\Phi)_1) = \top$ then
         return $\bot$
   return $\top$
else
   return $\bot$
```

Let us now assume $\psi = \langle \overline{B} \rangle \psi'$. $\texttt{Check}(\mathcal{K}, \psi, s, G, \mathcal{D}(\Phi)) = \top$ if and only if, for some $b'' \in \{1, \dots, |S| \cdot (2|\psi'| + 1) \cdot 2^{2\sum_{\ell=1}^{u} |r_\ell|} - 1\}$ and some $(G'', \mathcal{D}(\Phi)'', s'') \in \mathsf{Conf}(\mathcal{K}, \psi)$ $(= \mathsf{Conf}(\mathcal{K}, \psi'))$, we have $\texttt{Reach}(\mathcal{K}, \psi', (G, \mathcal{D}(\Phi), s), (G'', \mathcal{D}(\Phi)'', s''), b'') = \top$ and $\texttt{Check}(\mathcal{K}, \psi', s'', G'', \mathcal{D}(\Phi)'') = \top$. We preliminarily prove the following claim.

Claim 41. Let $b \in \mathbb{N}$, $b > 0$. Let $\tilde{\rho} \in \mathrm{Trc}_{\mathcal{K}}$ be a trace with $\tilde{s} = \mathrm{lst}(\tilde{\rho})$. Let $\tilde{G}$ be the subset of formulas in $\mathsf{Subf}_{\langle B \rangle}(\psi')$ that hold on some proper prefix of $\tilde{\rho}$. Let $\tilde{\mathcal{D}}(\Phi)$ be the current configuration of states of the DFAs associated with the regular expressions in Φ, reached from the initial states after reading $\mu(\tilde{\rho}(1, |\tilde{\rho}| - 1))$.

For $(\tilde{G}, \tilde{\mathcal{D}}(\Phi), \tilde{s})$, $(G', \mathcal{D}(\Phi)', s') \in \mathsf{Conf}(\mathcal{K}, \psi')$, we have that $\texttt{Reach}(\mathcal{K}, \psi', (\tilde{G}, \tilde{\mathcal{D}}(\Phi), \tilde{s}), (G', \mathcal{D}(\Phi)', s'), b) = \top$ if and only if there exists $\rho' \in \mathrm{Trc}_{\mathcal{K}}$ such that $\tilde{\rho} \cdot \rho' \in \mathrm{Trc}_{\mathcal{K}}$, $|\rho'| = b$, $\mathrm{lst}(\rho') = s'$, G' is the subset of formulas in $\mathsf{Subf}_{\langle B \rangle}(\psi')$ that hold on some proper prefix of $\tilde{\rho} \cdot \rho'$, and $\mathcal{D}(\Phi)'$ is the current configuration of the DFAs associated with the regular expressions of Φ, after reading $\mu(\tilde{\rho} \cdot \rho'(1, |\tilde{\rho} \cdot \rho'| - 1))$.

Proof. The proof is by induction on $b \geq 1$.

If $b = 1$, we have $\texttt{Reach}(\mathcal{K}, \psi', (\tilde{G}, \tilde{\mathcal{D}}(\Phi), \tilde{s}), (G', \mathcal{D}(\Phi)', s'), b) = \top$ iff $\texttt{Compatible}(\mathcal{K}, \psi', (\tilde{G}, \tilde{\mathcal{D}}(\Phi), \tilde{s}), (G', \mathcal{D}(\Phi)', s'))$ $= \top$. This happens if and only if:

1. $(\tilde{s}, s') \in R$, i.e., $(\tilde{s}, s')$ is an edge of $\mathcal{K}$;
2. $\texttt{advance}(\tilde{\mathcal{D}}(\Phi), \mu(\tilde{s})) = \mathcal{D}(\Phi)'$;
3. $\tilde{G} \subseteq G'$;
4. for each $\varphi \in (G' \setminus \tilde{G})$, $\texttt{Check}(\mathcal{K}, \varphi, \tilde{s}, \tilde{G} \cap \mathsf{Subf}_{\langle B \rangle}(\varphi), \tilde{\mathcal{D}}(\Phi)) = \top$;
5. for each $\varphi \in (\mathsf{Subf}_{\langle B \rangle}(\psi') \setminus G')$, $\texttt{Check}(\mathcal{K}, \varphi, \tilde{s}, \tilde{G} \cap \mathsf{Subf}_{\langle B \rangle}(\varphi), \tilde{\mathcal{D}}(\Phi)) = \bot$.

Let $\rho' = s'$. $(\Rightarrow)$ By the inductive hypothesis (of the external theorem over $\tilde{\rho}$), by point 4. it follows that $\mathcal{K}, \tilde{\rho} \models \varphi$ for each $\varphi \in (G' \setminus \tilde{G})$. By point 5. it follows that $\mathcal{K}, \tilde{\rho} \not\models \varphi$ for each $\varphi \in (\mathsf{Subf}_{\langle B \rangle}(\psi') \setminus G')$ and the claim follows.

Conversely, $(\Leftarrow)$ points 1., 2., and 3. easily follow. Moreover, it must hold that $\mathcal{K}, \tilde{\rho} \models \varphi$ for each $\varphi \in (G' \setminus \tilde{G})$, and $\mathcal{K}, \tilde{\rho} \not\models \varphi$ for each $\varphi \in (\mathsf{Subf}_{\langle B \rangle}(\psi') \setminus G')$ and, therefore, points 4. and 5. follow by the inductive hypothesis (of the external theorem).

If $b \geq 2$, $\texttt{Reach}(\mathcal{K}, \psi', (\tilde{G}, \tilde{\mathcal{D}}(\Phi), \tilde{s}), (G', \mathcal{D}(\Phi)', s'), b) = \top$ if and only if, for some $(G_3, \mathcal{D}(\Phi)_3, s_3) \in \mathsf{Conf}(\mathcal{K}, \psi')$, $\texttt{Reach}(\mathcal{K}, \psi', (\tilde{G}, \tilde{\mathcal{D}}(\Phi), \tilde{s}), (G_3, \mathcal{D}(\Phi)_3, s_3), \lfloor b/2 \rfloor) = \top$ and $\texttt{Reach}(\mathcal{K}, \psi', (G_3, \mathcal{D}(\Phi)_3, s_3), (G', \mathcal{D}(\Phi)', s'), b - \lfloor b/2 \rfloor) = \top$.

$(\Rightarrow)$ By the inductive hypothesis (over b), there exists $\rho_3 \in \mathrm{Trc}_{\mathcal{K}}$ such that $\tilde{\rho} \cdot \rho_3 \in \mathrm{Trc}_{\mathcal{K}}$, $|\rho_3| = \lfloor b/2 \rfloor$, $\mathrm{lst}(\rho_3) = s_3$, G_3 is the subset of subformulas in $\mathsf{Subf}_{\langle B \rangle}(\psi')$ that hold on some proper prefix of $\tilde{\rho} \cdot \rho_3$, and $\mathcal{D}(\Phi)_3$ is the current configuration of the DFAs associated with the regular expressions in Φ, after reading $\mu(\tilde{\rho} \cdot \rho_3(1, |\tilde{\rho} \cdot \rho_3| - 1))$.

By the inductive hypothesis (over b, applied to the trace $\tilde{\rho} \cdot \rho_3$), there exists $\rho' \in \mathrm{Trc}_{\mathcal{K}}$ such that $\tilde{\rho} \cdot \rho_3 \cdot \rho' \in \mathrm{Trc}_{\mathcal{K}}$, $|\rho'| = b - \lfloor b/2 \rfloor$, $\mathrm{lst}(\rho') = s'$, G' is the subset of subformulas in $\mathsf{Subf}_{\langle B \rangle}(\psi')$ that hold on some proper prefix of $\tilde{\rho} \cdot \rho_3 \cdot \rho'$, and $\mathcal{D}(\Phi)'$ is the current configuration of the DFAs associated with the regular expressions in Φ, after reading $\mu(\tilde{\rho} \cdot \rho_3 \cdot \rho'(1, |\tilde{\rho} \cdot \rho_3 \cdot \rho'| - 1))$. The claim follows, as $\rho_3 \cdot \rho' \in \mathrm{Trc}_{\mathcal{K}}$ and $|\rho_3 \cdot \rho'| = b$.

$(\Leftarrow)$ Conversely, there exists $\rho' \in \mathrm{Trc}_{\mathcal{K}}$ such that $\tilde{\rho} \cdot \rho' \in \mathrm{Trc}_{\mathcal{K}}$, $|\rho'| = b \geq 2$, $\mathrm{lst}(\rho') = s'$, G' is the subset of subformulas in $\mathsf{Subf}_{\langle B \rangle}(\psi')$ that hold on some proper prefix of $\tilde{\rho} \cdot \rho'$, and $\mathcal{D}(\Phi)'$ is the current configuration of the DFAs associated with the regular expressions in Φ, after reading $\mu(\tilde{\rho} \cdot \rho'(1, |\tilde{\rho} \cdot \rho'| - 1))$. Let us split $\rho' = \rho_3 \cdot \rho_4$, where $|\rho_3| = \lfloor b/2 \rfloor$ and $|\rho_4| = b - \lfloor b/2 \rfloor$. Let $(G_3, \mathcal{D}(\Phi)_3, s_3) \in \mathsf{Conf}(\mathcal{K}, \psi')$ be such that $\mathcal{D}(\Phi)_3$ is the current configuration of the DFAs associated with the regular expressions in Φ, after reading $\mu(\tilde{\rho} \cdot \rho_3(1, |\tilde{\rho} \cdot \rho_3| - 1))$, $s_3 = \mathrm{lst}(\rho_3)$, G_3 is the subset of subformulas in $\mathsf{Subf}_{\langle B \rangle}(\psi')$ that hold on some proper prefix of $\tilde{\rho} \cdot \rho_3$. By the inductive hypothesis (on b over $\tilde{\rho} \cdot \rho_3$), $\texttt{Reach}(\mathcal{K}, \psi', (G_3, \mathcal{D}(\Phi)_3, s_3), (G', \mathcal{D}(\Phi)', s'), b - \lfloor b/2 \rfloor) = \top$. Moreover, by the inductive hypothesis (on b over $\tilde{\rho}$), we have $\texttt{Reach}(\mathcal{K}, \psi', (\tilde{G}, \tilde{\mathcal{D}}(\Phi), \tilde{s}), (G_3, \mathcal{D}(\Phi)_3, s_3), \lfloor b/2 \rfloor) = \top$.

Hence, both the recursive calls at line 6 return $\top$, when at line 5 $(G_3, \mathcal{D}(\Phi)_3, s_3)$ is considered by the loop. Thus, $\texttt{Reach}(\mathcal{K}, \psi', (\tilde{G}, \tilde{\mathcal{D}}(\Phi), \tilde{s}), (G', \mathcal{D}(\Phi)', s'), b)$ returns $\top$ concluding the proof of the claim. □

($\Rightarrow$) Let us now assume that in the execution of the procedure `Check`, at lines 15–19, for some $b'' \in \{1, \dots, |S| \cdot (2|\psi'| + 1) \cdot 2^{2\sum_{\ell=1}^{u}|r_\ell|} - 1\}$ and some $(G'', \mathcal{D}(\Phi)'', s'') \in \mathsf{Conf}(\mathcal{K}, \psi)$ $(= \mathsf{Conf}(\mathcal{K}, \psi'))$, we have $\mathtt{Reach}(\mathcal{K}, \psi', (G, \mathcal{D}(\Phi), s), (G'', \mathcal{D}(\Phi)'', s''), b'') = \top$ and $\mathtt{Check}(\mathcal{K}, \psi', s'', G'', \mathcal{D}(\Phi)'') = \top$. By the claim above, there exists $\rho'' \in \mathrm{Trc}_{\mathcal{K}}$ such that $\rho \cdot \rho'' \in \mathrm{Trc}_{\mathcal{K}}$, $\mathsf{lst}(\rho'') = s''$, G'' is the subset of subformulas in $\mathsf{Subf}_{\langle \mathrm{B} \rangle}(\psi')$ that hold on some proper prefix of $\rho \cdot \rho''$, and $\mathcal{D}(\Phi)''$ is the current configuration of the DFAs associated with the regular expressions of Φ, after reading $\mu(\rho \cdot \rho''(1, |\rho \cdot \rho''| - 1))$. By the inductive hypothesis, since $\mathtt{Check}(\mathcal{K}, \psi', s'', G'', \mathcal{D}(\Phi)'') = \top$, we have $\mathcal{K}, \rho \cdot \rho'' \models \psi'$ implying that $\mathcal{K}, \rho \models \langle \overline{\mathrm{B}} \rangle\, \psi'$.

($\Leftarrow$) Conversely, if $\mathcal{K}, \rho \models \langle \overline{\mathrm{B}} \rangle\, \psi'$, we have $\mathcal{K}, \rho \cdot \rho'' \models \psi'$ for some $\rho'' \in \mathrm{Trc}_{\mathcal{K}}$, with $\rho \cdot \rho'' \in \mathrm{Trc}_{\mathcal{K}}$. By the exponential small-model property (Theorem 37), there exists $\rho' \in \mathrm{Trc}_{\mathcal{K}}$ such that $\mathsf{lst}(\rho'') = \mathsf{lst}(\rho')$, $|\rho'| \le |S| \cdot (2|\psi'| + 1) \cdot 2^{2\sum_{\ell=1}^{u}|r_\ell|} - 1$ (recall that the factor 2 in front of $|\psi'|$ is due to the fact that a formula in NNF is required), $\rho \cdot \rho' \in \mathrm{Trc}_{\mathcal{K}}$ and $\mathcal{K}, \rho \cdot \rho' \models \psi'$. Let G' be the subset of subformulas in $\mathsf{Subf}_{\langle \mathrm{B} \rangle}(\psi') = \mathsf{Subf}_{\langle \mathrm{B} \rangle}(\psi)$ that hold on some proper prefix of $\rho \cdot \rho'$, and $\mathcal{D}(\Phi)'$ be the current configuration of the DFAs associated with the regular expressions in Φ, after reading $\mu(\rho \cdot \rho'(1, |\rho \cdot \rho'| - 1))$. By the inductive hypothesis (over $\rho \cdot \rho'$), $\mathtt{Check}(\mathcal{K}, \psi', \mathsf{lst}(\rho'), G', \mathcal{D}(\Phi)') = \top$. By the claim above, $\mathtt{Reach}(\mathcal{K}, \psi', (G, \mathcal{D}(\Phi), s), (G', \mathcal{D}(\Phi)', \mathsf{lst}(\rho')), |\rho'|) = \top$, hence $\mathtt{Check}(\mathcal{K}, \psi, s, G, \mathcal{D}(\Phi)) = \top$. This concludes the proof of the theorem. □

Algorithm 6 reports the main MC procedure $\mathtt{CheckAux}(\mathcal{K}, \Phi)$ for B$\overline{\text{B}}$. It starts constructing the NFAs and the initial states of the DFAs for the regular expressions of Φ (line 1). Then, `CheckAux` invokes the procedure `Check` two times (line 2): the former to check the special case of the trace s_0 consisting of only the initial state of $\mathcal{K}$, and the latter for all right-extensions of s_0 (i.e., the initial traces having length at least 2). Notice that the NFAs and DFAs for the regular expressions of $\langle \overline{\mathrm{B}} \rangle \neg\Phi$, $\neg\Phi$ and Φ are the same (i.e. $\mathcal{D}(\Phi)_0 = \mathcal{D}(\langle \overline{\mathrm{B}} \rangle \neg\Phi)_0 = \mathcal{D}(\neg\Phi)_0$) allowing us to simultaneously apply the result of Theorem 40 for both the invocations of `Check` at line 2, proving soundness and completeness of the procedure.

Algorithm 6 $\mathtt{CheckAux}(\mathcal{K}, \Phi)$

```
1: create($\mathcal{D}(\Phi)_0$)        ▷ Creates the (succinct) NFAs and the initial states of the DFAs for all the RE in $\Phi$
2: if Check($\mathcal{K}, \neg\Phi, s_0, \emptyset, \mathcal{D}(\Phi)_0$) or Check($\mathcal{K}, \langle\overline{\mathrm{B}}\rangle\neg\Phi, s_0, \emptyset, \mathcal{D}(\Phi)_0$) then
3:     return $\bot$
4: else
5:     return $\top$
```

Theorem 42. *Let $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ be a finite Kripke structure, and Φ be a* B$\overline{\text{B}}$ *formula. Then,* $\mathtt{CheckAux}(\mathcal{K}, \Phi) = \top \iff \mathcal{K} \models \Phi$.

Proof. If $\mathcal{K} \models \Phi$, then for all $\rho \in \mathrm{Trc}_{\mathcal{K}}$ with $\mathsf{fst}(\rho) = s_0$, we have $\mathcal{K}, \rho \models \Phi$. Hence, we have $\mathcal{K}, s_0 \models \Phi$, and $\mathcal{K}, s_0 \cdot \rho' \models \Phi$ for all $s_0 \cdot \rho' \in \mathrm{Trc}_{\mathcal{K}}$, implying that $\mathcal{K}, s_0 \models [\overline{B}]\Phi$ and $\mathcal{K}, s_0 \not\models \langle \overline{\mathrm{B}} \rangle \neg\Phi$. By Theorem 40, $\mathtt{Check}(\mathcal{K}, \neg\Phi, s_0, \emptyset, \mathcal{D}(\Phi)_0) = \bot$ and $\mathtt{Check}(\mathcal{K}, \langle \overline{\mathrm{B}} \rangle \neg\Phi, s_0, \emptyset, \mathcal{D}(\Phi)_0) = \bot$ implying that $\mathtt{CheckAux}(\mathcal{K}, \Phi) = \top$. Conversely, if $\mathtt{CheckAux}(\mathcal{K}, \Phi) = \top$, then it must be $\mathtt{Check}(\mathcal{K}, \neg\Phi, s_0, \emptyset, \mathcal{D}(\Phi)_0) = \bot$ and $\mathtt{Check}(\mathcal{K}, \langle \overline{\mathrm{B}} \rangle \neg\Phi, s_0, \emptyset, \mathcal{D}(\Phi)_0) = \bot$. By Theorem 40 applied to the trace $\rho = s_0$, we have $\mathcal{K}, s_0 \not\models \neg\Phi$ and $\mathcal{K}, s_0 \not\models \langle \overline{\mathrm{B}} \rangle \neg\Phi$, and thus $\mathcal{K} \models \Phi$. □

The following corollary states the upper bound to the complexity of MC for B$\overline{\text{B}}$.

Corollary 43. *The MC problem for* B$\overline{\text{B}}$ *formulas on finite Kripke structures is in* **PSPACE***.*

Proof. The procedure `CheckAux` decides the problem using *polynomial working space* basically due to two facts. The first one is the number of simultaneously active recursive calls of `Check`, which is $O(|\Phi|)$. The second is the space (in bits) used for any call of `Check`, that is,

$$O\Big(|\Phi| + |S| + \sum_{\ell=1}^{u} |r_\ell| + \underbrace{\log(|S| \cdot |\Phi| \cdot 2^{2\sum_{\ell=1}^{u}|r_\ell|})}_{(1)} + \underbrace{(|\Phi| + |S| + \sum_{\ell=1}^{u} |r_\ell|)}_{(2)} \cdot \underbrace{\log(|S| \cdot |\Phi| \cdot 2^{2\sum_{\ell=1}^{u}|r_\ell|})}_{(3)}\Big).$$

In particular, (1) $O(\log(|S| \cdot |\Phi| \cdot 2^{2\sum_{\ell=1}^{u}|r_\ell|}))$ bits are used for the bound b on the trace length, (3) for *each subformula* $\langle \overline{\mathrm{B}} \rangle\, \psi'$ of Φ at most $O(\log(|S| \cdot |\Phi| \cdot 2^{2\sum_{\ell=1}^{u}|r_\ell|}))$ recursive calls of `Reach` may be simultaneously active (the recursion depth of `Reach` is logarithmic in b), and (2) each call of `Reach` requires $O(|\Phi| + |S| + \sum_{\ell=1}^{u} |r_\ell|)$ bits. □

Finally, since a Kripke structure can be unraveled against the direction of its edges, and a language $\mathcal{L}$ is regular if and only if its reversed version $\mathcal{L}^{\mathrm{Rev}} = \{w(|w|) \cdots w(1) \mid w \in \mathcal{L}\}$ is, the proposed algorithm can be easily modified to deal with the symmetric fragment E$\overline{\text{E}}$ proving that also the MC problem for E$\overline{\text{E}}$ is in **PSPACE**.

6.3. **PSPACE**-*completeness of MC for* $\mathsf{A\overline{A}B\overline{B}}$

In this section, we show that the algorithm `CheckAux` can be used as a basic engine to design a **PSPACE** MC algorithm for the bigger fragment $\mathsf{A\overline{A}B\overline{B}}$.

The idea is that, being the proposition letters related with regular expressions, the modalities $\langle \mathsf{A}\rangle$ and $\langle \overline{\mathsf{A}}\rangle$ do not augment the expressiveness of the fragment $\mathsf{B\overline{B}}$. In particular, we will show that the occurrences of modalities $\langle \mathsf{A}\rangle$ and $\langle \overline{\mathsf{A}}\rangle$ in an $\mathsf{A\overline{A}B\overline{B}}$ formula can suitably be "absorbed" and replaced by fresh proposition letters. We recall that $\mathcal{K}, \rho \models \langle \mathsf{A}\rangle\,\psi$ if and only if there exists a trace $\tilde{\rho} \in \mathrm{Trc}_{\mathcal{K}}$ such that $\mathrm{lst}(\rho) = \mathrm{fst}(\tilde{\rho})$ and $\mathcal{K}, \tilde{\rho} \models \psi$. An immediate consequence is that, for any $\rho' \in \mathrm{Trc}_{\mathcal{K}}$ with $\mathrm{lst}(\rho) = \mathrm{lst}(\rho')$, $\mathcal{K}, \rho \models \langle \mathsf{A}\rangle\,\psi \iff \mathcal{K}, \rho' \models \langle \mathsf{A}\rangle\,\psi$ and similarly for the symmetrical modality $\langle \overline{\mathsf{A}}\rangle$ with respect to the first state of the trace. In general, if two traces have the same final state (respectively, first state), either both of them satisfy a formula $\langle \mathsf{A}\rangle\,\psi$ (respectively, $\langle \mathsf{A}\rangle\,\psi$), or none of them does. As a consequence, for a formula $\langle \mathsf{A}\rangle\,\psi$ (respectively, $\langle \overline{\mathsf{A}}\rangle\,\psi$), we can determine the subset $S_{\langle \mathsf{A}\rangle\,\psi}$ (respectively, $S_{\langle \overline{\mathsf{A}}\rangle\,\psi}$) of the set of states S of the Kripke structure such that, for any $\rho \in \mathrm{Trc}_{\mathcal{K}}$, $\mathcal{K}, \rho \models \langle \mathsf{A}\rangle\,\psi$ (respectively, $\mathcal{K}, \rho \models \langle \overline{\mathsf{A}}\rangle\,\psi$) if and only if $\mathrm{lst}(\rho) \in S_{\langle \mathsf{A}\rangle\,\psi}$ (respectively, $\mathrm{fst}(\rho) \in S_{\langle \overline{\mathsf{A}}\rangle\,\psi}$).

Now, for a formula $\langle \mathsf{A}\rangle\,\psi$ (respectively, $\langle \overline{\mathsf{A}}\rangle\,\psi$), we provide a regular expression $r_{\langle \mathsf{A}\rangle\,\psi}$ (respectively, $r_{\langle \overline{\mathsf{A}}\rangle\,\psi}$) characterizing the set of traces which model the formula. To this end we identify states in S by a set of fresh proposition letters $\{q_s \mid s \in S\}$ and we replace the Kripke structure $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ by $\mathcal{K}' = (\mathcal{AP}', S, R, \mu', s_0)$, with $\mathcal{AP}' := \mathcal{AP} \cup \{q_s \mid s \in S\}$ and $\mu'(s) = \{q_s\} \cup \mu(s)$ for any $s \in S$. The regular expressions $r_{\langle \mathsf{A}\rangle\,\psi}$ and $r_{\langle \overline{\mathsf{A}}\rangle\,\psi}$ are

$$r_{\langle \mathsf{A}\rangle\,\psi} := \top^* \cdot \Big(\bigcup_{s \in S_{\langle \mathsf{A}\rangle\,\psi}} q_s \Big) \quad \text{and} \quad r_{\langle \overline{\mathsf{A}}\rangle\,\psi} := \Big(\bigcup_{s \in S_{\langle \overline{\mathsf{A}}\rangle\,\psi}} q_s \Big) \cdot \top^*.$$

By definition $\mathcal{K}, \rho \models r_{\langle \mathsf{A}\rangle\,\psi}$ if and only if $\mathrm{lst}(\rho) \in S_{\langle \mathsf{A}\rangle\,\psi}$, if and only if $\mathcal{K}, \rho \models \langle \mathsf{A}\rangle\,\psi$.

We can now sketch the procedure for "reducing" the MC problem for $\mathsf{A\overline{A}B\overline{B}}$ to the MC problem for $\mathsf{B\overline{B}}$. We iteratively rewrite a formula Φ of $\mathsf{A\overline{A}B\overline{B}}$ until it gets converted to an (equivalent) formula of $\mathsf{B\overline{B}}$. At each step, we select an occurrence of a subformula of Φ, either having the form $\langle \mathsf{A}\rangle\,\psi$ or $\langle \overline{\mathsf{A}}\rangle\,\psi$, devoid of any occurrence of modalities $\langle \mathsf{A}\rangle$ and $\langle \overline{\mathsf{A}}\rangle$ in ψ. For such an occurrence, say $\langle \mathsf{A}\rangle\,\psi$, we have to compute the set $S_{\langle \mathsf{A}\rangle\,\psi}$. For that purpose we can run a variant `CheckAux'`$(\mathcal{K}, \Psi, s)$ of the MC procedure `CheckAux`$(\mathcal{K}, \Psi)$, which invokes `Check` at line 2 on the additional parameter (state) s, instead of s_0. For each $s \in S$, we invoke `CheckAux'`$(\mathcal{K}, \neg\psi, s)$, deciding that $s \in S_{\langle \mathsf{A}\rangle\,\psi}$ if and only if the procedure returns $\bot$. Then we replace $\langle \mathsf{A}\rangle\,\psi$ in Φ with (a fresh proposition letter associated with) the regular expression $r_{\langle \mathsf{A}\rangle\,\psi}$, obtaining a formula Φ'. To deal with subformulas of the form $\langle \overline{\mathsf{A}}\rangle\,\psi$, we have to introduce a slight variant of the procedure `Check`, which finds traces leading to (and not starting from) a given state. Now, if the resulting formula Φ' is in $\mathsf{B\overline{B}}$, the rewriting ends; otherwise, we can perform another rewriting step over Φ'.

Considering that the sets $S_{\langle \mathsf{A}\rangle\,\psi}$, $S_{\langle \overline{\mathsf{A}}\rangle\,\psi'}$ and the regular expressions $r_{\langle \mathsf{A}\rangle\,\psi}$ and $r_{\langle \overline{\mathsf{A}}\rangle\,\psi}$ have a size linear in $|S|$, we can conclude with the following result.

Theorem 44. *The MC problem for* $\mathsf{A\overline{A}B\overline{B}}$ *over finite Kripke structures is in* **PSPACE**.

By symmetry we can show that the MC problem for $\mathsf{A\overline{A}E\overline{E}}$ formulas is also in **PSPACE**.

The **PSPACE**-hardness of MC for $\mathsf{B\overline{B}}$ and $\mathsf{A\overline{A}B\overline{B}}$ directly follows from that of the smallest fragment Prop (the purely propositional fragment of HS) which is stated by Theorem 45. As a matter of fact, in A.4 we prove that Prop is hard for **PSPACE** by a reduction from the **PSPACE**-complete *universality problem for regular expressions* [17] (the problem of deciding, for a regular expression r with $\mathcal{L}(r) \subseteq \Sigma^*$ and $|\Sigma| \geq 2$, whether $\mathcal{L}(r) = \Sigma^*$).

Theorem 45. *The MC problem for* Prop *formulas over finite Kripke structures is* **PSPACE**-*hard (under* **LOGSPACE** *reductions).*

By Theorem 44 and Theorem 45 we obtain the following complexity result.

Theorem 46. *The MC problem for formulas of any (proper or improper) sub-fragment of* $\mathsf{A\overline{A}B\overline{B}}$ *(and* $\mathsf{A\overline{A}E\overline{E}}$*) over finite Kripke structures is* **PSPACE**-*complete.*

7. Conclusions

In this paper, we have studied the MC problem for HS extended with regular expressions used to define interval labeling. The approach, stemming from [25], generalizes both the one of [29] that enforces the homogeneity principle and of [23,24] where labeling is endpoint-based. In the general case, MC problem for (full) HS turns out to be nonelementarily decidable—the proof exploits an automata-theoretic approach based on the notion of $\mathcal{K}$-NFA— but, for a constant-length formula, it is in **P**. Moreover, the MC problem is **EXPSPACE**-hard (the hardness follows from that of BE under homogeneity [6]).

Moreover, we have investigated the MC problem for two maximal fragments of HS, namely $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}E\overline{B}\overline{E}}$ with regular expressions, and we have showed that it is $\mathbf{AEXP_{pol}}$-complete. The complexity upper bound has been proved by providing an alternating algorithm which performs an exponential number of computation steps, but only polynomially many alternations (in the length of the formula to be checked). Conversely, the lower bound has been shown by a reduction from the $\mathbf{AEXP_{pol}}$-complete "alternating multi-tiling problem". In this way, we have also improved the known complexity result for the same fragments under the homogeneity assumption.

Finally, we have proved that the HS fragments, $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$ and $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{E}}$ and all their sub-fragments are **PSPACE**-complete. In fact we have shown that $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$ and $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{E}}$ enjoy a suitable small-model property that allows us to check formulas of $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}/\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{E}}$ only against traces having at most exponential length. Conversely, the matching complexity lower bound has been proved by a reduction from the **PSPACE**-complete universality problem for regular expressions.

Future work will focus on the problem of determining the exact complexity of MC for full HS, both under homogeneity and in the regular expression-based semantics. In addition, we will study MC for HS over *visibly pushdown systems* (VPS), that allow us to deal with recursive programs and infinite state systems. Finally, we are thinking of inherently *interval-based models of systems*: Kripke structures, being based on states, are naturally oriented to the description of point-based properties of systems, and of how they evolve state-by-state. We want to come up with suitable (and practical) description paradigms for systems, which should enable us to directly model them on the basis of their interval behavior/properties. Only after devising these models (something that seems to be extremely challenging!) a really general interval-based MC will be possible.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Proofs and complements

A.1. Completion of the proof of Proposition 7

Construction for the language $\langle \mathrm{E}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$. Let us consider the NFA $\mathcal{A}_{\langle \mathrm{E}\rangle}$ over S given by $\mathcal{A}_{\langle \mathrm{E}\rangle} = (S, (M \cup \{q_0'\}) \times S, \{q_0'\} \times S, \delta', F)$, where $q_0' \notin M$ is a fresh main state and for all $(q, s) \in (M \cup \{q_0'\}) \times S$ and $s' \in S$, we have $\delta'((q, s), s') = \emptyset$, if $s' \neq s$, and $\delta((q, s), s)$ is defined as follows:

$$\delta((q,s),s) = \begin{cases} \delta((q,s),s) & \text{if } q \neq q_0' \\ (\{q_0'\} \times R(s)) \cup \{(q_0, s') \in Q_0 \mid s' \in R(s)\} & \text{otherwise.} \end{cases}$$

Starting from an initial state (q_0', s), the automaton $\mathcal{A}_{\langle \mathrm{E}\rangle}$ either remains in a state whose main component is q_0', or moves to an initial state (q_0, s') of $\mathcal{A}$, ensuring at the same time that the portion of the input read so far is faithful to the evolution of $\mathcal{K}$. From the state (q_0, s'), $\mathcal{A}_{\langle \mathrm{E}\rangle}$ simulates the behavior of $\mathcal{A}$. Formally, since $\mathcal{A}$ is a $\mathcal{K}$-NFA, by construction it easily follows that $\mathcal{A}_{\langle \mathrm{E}\rangle}$ is a $\mathcal{K}$-NFA which accepts the set of traces of $\mathcal{K}$ having a non-empty proper suffix in $\mathcal{L}(\mathcal{A})$. Hence, $\mathcal{L}(\mathcal{A}_{\langle \mathrm{E}\rangle}) = \langle \mathrm{E}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$.

Construction for the language $\langle \overline{\mathrm{E}}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$. Let us consider the NFA $\mathcal{A}_{\langle \overline{\mathrm{E}}\rangle}$ over S given by $\mathcal{A}_{\langle \overline{\mathrm{E}}\rangle} = (S, (M \cup \{q_{acc}\}) \times S, Q_0', \delta', \{q_{acc}\} \times S)$, where $q_{acc} \notin M$ is a fresh main state, and Q_0' and δ' are defined as follows:

- the set Q_0' of initial states is the set of states (q, s) of $\mathcal{A}$ such that there is a run of $\mathcal{A}$ from some initial state to (q, s) over some non-empty word.
- For all $(q, s) \in (M \cup \{q_{acc}\}) \times S$ and $s' \in S$, we have $\delta'((q, s), s') = \emptyset$, if $s' \neq s$, and

$$\delta'((q,s),s) = \begin{cases} \delta((q,s),s) \cup \displaystyle\bigcup_{(q',s') \in F \cap \delta((q,s),s)} \{(q_{acc}, s')\} & \text{if } q \in M \\ \emptyset & \text{if } q = q_{acc}. \end{cases}$$

Note that the set Q_0' can be computed in time polynomial in the size of $\mathcal{A}$. Since $\mathcal{A}_{\langle \overline{\mathrm{E}}\rangle}$ essentially simulates $\mathcal{A}$, and $\mathcal{A}$ is a $\mathcal{K}$-NFA, by construction we easily obtain that $\mathcal{A}_{\langle \overline{\mathrm{E}}\rangle}$ is a $\mathcal{K}$-NFA which accepts the set of words over S which are non-empty proper suffixes of words in $\mathcal{L}(\mathcal{A})$. Thus, since $\mathcal{A}$ is a $\mathcal{K}$-NFA, we obtain that $\mathcal{L}(\mathcal{A}_{\langle \overline{\mathrm{E}}\rangle}) = \langle \overline{\mathrm{E}}\rangle_{\mathcal{K}}(\mathcal{L}(\mathcal{A}))$. □

A.2. Pseudocode of `checkFalse`

Algorithm 7 $\texttt{checkFalse}_{(\mathcal{K},\varphi,Lab)}(\mathcal{W})$ [*$\mathcal{W}$: well-formed set, Lab: $\mathsf{A}\overline{\mathsf{A}}$-labeling for $(\mathcal{K},\varphi)$*]

```
while $\mathcal{W}$ is not universal do
    deterministically select $(\psi,\rho)\in\mathcal{W}$ such that $\psi$ does not have the form $[\overline{E}]\psi'$ and $[\overline{B}]\psi'$
    $\mathcal{W}\leftarrow\mathcal{W}\setminus\{(\psi,\rho)\}$
    Case $\psi=r$ with $r\in\mathsf{RE}$
        if $\rho\notin\mathcal{L}(r)$ then
            accept the input
    case $\psi=\neg r$ with $r\in\mathsf{RE}$
        if $\rho\in\mathcal{L}(r)$ then
            accept the input
    case $\psi=\langle\mathrm{A}\rangle\,\psi'$ or $\psi=[A]\psi'$
        if $\psi\notin Lab(\mathrm{lst}(\rho))$ then
            accept the input
    case $\psi=\langle\overline{\mathrm{A}}\rangle\,\psi'$ or $\psi=[\overline{A}]\psi'$
        if $\psi\notin Lab(\mathrm{fst}(\rho))$ then
            accept the input
    case $\psi=\psi_1\vee\psi_2$
        universally choose $i=1,2$
        $\mathcal{W}\leftarrow\mathcal{W}\cup\{(\psi_i,\rho)\}$
    case $\psi=\psi_1\wedge\psi_2$
        $\mathcal{W}\leftarrow\mathcal{W}\cup\{(\psi_1,\rho),(\psi_2,\rho)\}$
    case $\psi=\langle\mathrm{B}\rangle\,\psi'$
        universally choose $\rho'\in\mathrm{Pref}(\rho)$
        $\mathcal{W}\leftarrow\mathcal{W}\cup\{(\psi',\rho')\}$
    case $\psi=[B]\psi'$
        $\mathcal{W}\leftarrow\mathcal{W}\cup\{(\psi',\rho')\mid\rho'\in\mathrm{Pref}(\rho)\}$
    case $\psi=\langle X\rangle\psi'$ with $X\in\{\overline{E},\overline{B}\}$
        universally choose an $X$-witness $\rho'$ of $\rho$ for $(\mathcal{K},\varphi)$
        $\mathcal{W}\leftarrow\mathcal{W}\cup\{(\psi',\rho')\}$
    EndCase
if $\mathcal{W}=\emptyset$ then
    reject the input
else
    existentially choose $(\psi,\rho)\in\widetilde{\mathcal{W}}$
    $\texttt{checkTrue}_{(\mathcal{K},\varphi,Lab)}(\{(\psi,\rho)\})$
```

A.3. Proof of Proposition 26

For technical convenience, in order to prove Proposition 26, for an $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}\overline{\mathsf{E}}$ formula φ we consider a slight variant $\Upsilon_w(\varphi)$ of $\Upsilon(\varphi)$. Formally, $\Upsilon_w(\varphi)$ is given by $\Upsilon(\langle\overline{\mathrm{B}}\rangle\,\varphi)$ (or, equivalently, by $\Upsilon(\langle\overline{\mathrm{E}}\rangle\,\varphi)$). Note that for each $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}\overline{\mathsf{E}}$ formula φ and $X\in\{\overline{E},\overline{B}\}$, we have $\Upsilon_w([X]\varphi)=\Upsilon_w(\widetilde{[X]\varphi})+1$.

Let $\mathcal{K}$ be a finite Kripke structure, φ be an $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}\overline{\mathsf{E}}$ formula in NNF, and $\mathcal{W}$ be a well-formed set for $(\mathcal{K},\varphi)$. We denote by $\Upsilon_w(\mathcal{W})$ the maximum over the alternation depths $\Upsilon_w(\psi)$, where ψ is a formula occurring in $\mathcal{W}$ (we set $\Upsilon_w(\mathcal{W})=0$ if $\mathcal{W}=\emptyset$). For each non-empty *universal* well-formed set $\mathcal{W}$ for $(\mathcal{K},\varphi)$, we have $\Upsilon_w(\widetilde{\mathcal{W}})=\Upsilon_w(\mathcal{W})-1$.

Now, we can prove Proposition 26.

Proposition *(26). The ATM `check` is a singly exponential-time bounded ATM accepting* FMC *whose number of alternations on input $(\mathcal{K},\varphi)$ is at most $\Upsilon(\varphi)+2$.*

Proof. Let us fix an input $(\mathcal{K},\varphi)$, where φ is an $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}\overline{\mathsf{E}}$ formula in NNF.

Note that whenever there is a switch between the procedures `checkTrue` and `checkFalse`, e.g., from `checkTrue` to `checkFalse`, (*i*) the input $\{(\psi,\rho)\}$ of the called procedure is contained in the dual $\widetilde{\mathcal{W}}$ of the currently processed well-formed set $\mathcal{W}$ for $(\mathcal{K},\varphi)$, and (*ii*) $\mathcal{W}$ is non-empty and universal: hence $\Upsilon_w(\{(\psi,\rho)\})<\Upsilon_w(\mathcal{W})$. Moreover, a well-formed set $\mathcal{W}$ for $(\mathcal{K},\varphi)$ contains only formulas ψ such that $\psi\in\mathsf{SD}(\varphi)$.

Additionally, in each iteration of the while loops of procedures `checkTrue` and `checkFalse`, the processed pair (ψ,ρ) in the current well-formed set $\mathcal{W}$ is either removed from $\mathcal{W}$, or it is replaced with pairs (ψ',ρ') such that ψ' is a strict subformula of ψ. This ensures that the algorithm always terminates.

Furthermore, since the number of alternations of the ATM `check` between existential choices and universal choices is evidently the number of switches between the calls to procedures `checkTrue` and `checkFalse` plus 2, and the top calls to `checkTrue` take as input well-formed sets for $(\mathcal{K},\varphi)$ having the form $\{(\psi,\rho)\}$ where $\psi\in\mathsf{SD}(\varphi)$, we have proved the following result.

Claim 47. The number of alternations of the ATM `check` on input $(\mathcal{K},\varphi)$ is at most $\Upsilon(\varphi)+2$.

Next, we prove the following property.

Claim 48. The ATM `check` runs in time singly exponential in the size of the input.

Proof. Let us fix an input $(\mathcal{K},\varphi)$. Let $T(\varphi)$ be the standard tree encoding of φ, where each node is labeled by some subformula of φ. Let $\psi\in\mathsf{SD}(\varphi)$. If ψ is a subformula of φ, we define d_ψ as the maximum over the distances from the root in $T(\varphi)$ of ψ-labeled nodes. If, conversely, ψ is the dual of a subformula of φ, we let $d_\psi:=d_{\tilde{\psi}}$.

Let us denote by $H(\mathcal{K},\varphi)$ the length of a certificate for $(\mathcal{K},\varphi)$. Recall that $H(\mathcal{K},\varphi)=(|S|\cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$, where S is the set of states of $\mathcal{K}$, spec is the set of atomic formulas (regular expressions) occurring in φ, and $h=\mathsf{d}_{\mathsf{B}}(\varphi)$.

By Proposition 25, it follows that each step in an iteration of the while loops in the procedures `checkTrue` and `checkFalse` can be performed in time singly exponential in the size of $(\mathcal{K},\varphi)$. Thus, in order to prove Claim 48, it suffices to show that for all computations π of the ATM `check` starting from the input $(\mathcal{K},\varphi)$, the overall number N_ψ of iterations of the while loops (of procedures `checkTrue` and `checkFalse`) along π, where the formula ψ is processed, is at most $(2^{|\varphi|}\cdot H(\mathcal{K},\varphi))^{d_\psi}$.

The proof is done by induction on d_ψ. As for the base case, we have $d_\psi=0$. Therefore, $\psi=\varphi$ or $\psi=\widetilde{\varphi}$; by construction of the algorithm, N_φ and $N_{\widetilde{\varphi}}$ are at most equal to 1. Thus, the result holds.

As for the inductive step, let us assume that $d_\psi>0$. We consider the case where ψ is a subformula of φ (the case where $\widetilde{\psi}$ is a subformula of φ is similar). Then, the result follows from the following chain of inequalities, where $P(\psi)$ denotes the set of nodes of $T(\varphi)$ which are parents of the nodes labeled by ψ, and for each node x, $fo(x)$ denotes the formula labeling x.

$$N_\psi \le \sum_{x\in P(\psi)} N_{fo(x)}\cdot H(\mathcal{K},\varphi) \le \sum_{x\in P(\psi)} (2^{|\varphi|}\cdot H(\mathcal{K},\varphi))^{d_{fo(x)}}\cdot H(\mathcal{K},\varphi) \le \left(2^{|\varphi|}\cdot H(\mathcal{K},\varphi)\right)^{d_\psi}$$

The first inequality directly follows from the construction of the algorithm (note that if $fo(x)=[B]\psi$ the processing of the subformula $fo(x)$ in an iteration of the two while loops generates at most $H(\mathcal{K},\varphi)$ new “copies” of ψ). The second inequality follows by the inductive hypothesis, and the last one from the fact that $|P(\psi)|\le 2^{|\varphi|}$ and $d_{fo(x)}\le d_\psi-1$ for all $x\in P(\psi)$. This concludes the proof of Claim 48. □

It remains to show that the ATM `check` accepts FMC. Let us fix an input $(\mathcal{K},\varphi)$ and let *Lab* be the $\mathsf{A\overline{A}}$-labeling initially and existentially guessed by `check` (at line 1). Evidently, after the top calls to `checkTrue`, each configuration of the procedure `check` can be described by a tuple $(\ell, Lab, \mathcal{W}, f)$, where:

- $\mathcal{W}$ is a well-formed set for $(\mathcal{K},\varphi)$,
- $f=$ `true` if $\mathcal{W}$ is processed within `checkTrue`, and $f=$ `false` otherwise, and
- ℓ is an instruction label corresponding to one of the instructions of the procedures `checkTrue` and `checkFalse`.

We denote by ℓ_0 the label associated with the while instruction. A *main configuration* is a configuration having label ℓ_0.

Let $Lab_{\mathcal{W}}$ be the restriction of *Lab* to the set of formulas in $\mathsf{A\overline{A}}(\varphi)$ which are subformulas of formulas occurring in $\mathcal{W}$. In other words, for each state s, $Lab_{\mathcal{W}}(s)$ contains all and only the formulas $\psi\in Lab(s)$ such that either ψ or its dual $\widetilde{\psi}$ is a subformula of some formula occurring in $\mathcal{W}$. $Lab_{\mathcal{W}}$ is said to be *valid* if for all states s and $\psi\in Lab_{\mathcal{W}}(s)$, it holds $\mathcal{K},s\models\psi$.

Claim 49. Let $\mathcal{W}$ be a well-formed set for $(\mathcal{K},\varphi)$ and let us assume that $Lab_{\mathcal{W}}$ is valid. Then:

1. the main configuration $(\ell_0, Lab, \mathcal{W}, \texttt{true})$ leads to acceptance *if and only if* $\mathcal{W}$ is valid;
2. the main configuration $(\ell_0, Lab, \mathcal{W}, \texttt{false})$ leads to acceptance *if and only if* $\mathcal{W}$ is *not* valid.

Proof. We associate with $\mathcal{W}$ a natural number $\|\mathcal{W}\|$ defined as follows. Let us fix an ordering $\psi_1,\ldots,\psi_k$ of the formulas in $\mathsf{SD}(\varphi)$ such that, for all $i\neq j$, $|\psi_i|>|\psi_j|$ implies $i<j$.

First, we associate with $\mathcal{W}$ a $(k+1)$-tuple $(n_0,n_1,\ldots,n_k)$ of natural numbers defined as: the first component n_0 in the tuple is the alternation depth $\Upsilon_w(\mathcal{W})$ and, for all the other components n_i, with $1\le i\le k$, n_i is the number of elements of $\mathcal{W}$ associated with the formula ψ_i (i.e., the number of elements having the form (ψ_i,ρ)).

Then, $\|\mathcal{W}\|$ is the position of the tuple $(n_0,n_1,\ldots,n_k)$ along the total lexicographic ordering over $\mathbb{N}^{k+1}$. Note that if $\mathcal{W}$ is non-empty and universal, since $\Upsilon_w(\widetilde{\mathcal{W}})<\Upsilon_w(\mathcal{W})$, it holds that $\|\widetilde{\mathcal{W}}\|<\|\mathcal{W}\|$. Moreover, $\|\mathcal{W}\|$ strictly decreases at each iteration of the while loops in the procedures `checkTrue` and `checkFalse` (this is because at each iteration $\Upsilon_w(\mathcal{W})$ does not increase, and an element of $\mathcal{W}$ is replaced with elements associated with smaller formulas).

The proof of Claim 49 is now carried out by induction on $\|\mathcal{W}\|$. As for the base case we have $\|\mathcal{W}\|=0$, thus $\mathcal{W}$ is empty and clearly valid. By construction `checkTrue` accepts the empty set, while `checkFalse` rejects the empty set. The result holds.

As for the inductive step, let $\|\mathcal{W}\|>0$, hence $\mathcal{W}$ is not empty. First, assume that $\mathcal{W}$ is universal. Recall that $\|\widetilde{\mathcal{W}}\|<\|\mathcal{W}\|$. Thus:

- (1.) $\mathcal{W}$ is valid $\Longleftrightarrow$ for each $(\psi,\rho)\in\widetilde{\mathcal{W}}$, $\{(\psi,\rho)\}$ is not valid $\Longleftrightarrow$ (by the inductive hypothesis) for each $(\psi,\rho)\in\widetilde{\mathcal{W}}$, the main configuration $(\ell_0, Lab, \{(\psi,\rho)\}, \texttt{false})$ leads to acceptance $\Longleftrightarrow$ (by construction of the algorithm and since $\mathcal{W}$ is universal) the main configuration $(\ell_0, Lab, \mathcal{W}, \texttt{true})$ leads to acceptance.
- (2.) $\mathcal{W}$ is *not* valid $\Longleftrightarrow$ for some $(\psi,\rho)\in\widetilde{\mathcal{W}}$, $\{(\psi,\rho)\}$ is valid $\Longleftrightarrow$ (by the inductive hypothesis) for some $(\psi,\rho)\in\widetilde{\mathcal{W}}$, the main configuration $(\ell_0, Lab, \{(\psi,\rho)\}, \texttt{true})$ leads to acceptance $\Longleftrightarrow$ (by construction of the algorithm and since $\mathcal{W}$ is universal) the main configuration $(\ell_0, Lab, \mathcal{W}, \texttt{false})$ leads to acceptance.

Hence, (1.) and (2.) of Claim 49 hold if $\mathcal{W}$ is universal.

Now, let us assume that the non-empty set $\mathcal{W}$ is not universal. We consider (2.) of Claim 49 (the proof of (1.) is just the "dual"). Let $(\psi, \rho) \in \mathcal{W}$ be the pair selected by the procedure `checkFalse` in the iteration of the while loop associated with the main configuration $(\ell_0, \mathit{Lab}, \mathcal{W}, \texttt{false})$. Here we examine the cases where either $\psi = \langle \mathrm{A} \rangle \psi'$, or $\psi = [B]\psi'$, or $\psi = \langle X \rangle \psi'$ with $X \in \{\overline{B}, \overline{E}\}$ (the other cases are similar or simpler).

- $\psi = \langle \mathrm{A} \rangle \psi'$. We have that $\{(\langle \mathrm{A} \rangle \psi', \rho)\}$ is valid if and only if $\mathcal{K}, \mathrm{lst}(\rho) \models \langle \mathrm{A} \rangle \psi'$. By hypothesis, $\mathit{Lab}_{\mathcal{W}}$ is valid. Hence $\{(\langle \mathrm{A} \rangle \psi', \rho)\}$ is not valid if and only if $\langle \mathrm{A} \rangle \psi' \notin \mathit{Lab}_{\mathcal{W}}(\mathrm{lst}(\rho))$. Let $\mathcal{W}' = \mathcal{W} \setminus \{(\psi, \rho)\}$. Note that $\|\mathcal{W}'\| < \|\mathcal{W}\|$. Then $\mathcal{W}$ is not valid $\iff$ either $\langle \mathrm{A} \rangle \psi' \notin \mathit{Lab}_{\mathcal{W}}(\mathrm{lst}(\rho))$ or $\mathcal{W}'$ is not valid $\iff$ (by the inductive hypothesis) either $\langle \mathrm{A} \rangle \psi' \notin \mathit{Lab}_{\mathcal{W}}(\mathrm{lst}(\rho))$ or the main configuration $(\ell_0, \mathit{Lab}, \mathcal{W}', \texttt{false})$ leads to acceptance $\iff$ (by construction of `checkFalse`) the main configuration $(\ell_0, \mathit{Lab}, \mathcal{W}, \texttt{false})$ leads to acceptance.
- $\psi = [B]\psi'$. Let $\mathcal{W}' = (\mathcal{W} \setminus \{(\psi, \rho)\}) \cup \{(\psi', \rho') \mid \rho' \in \mathrm{Pref}(\rho)\}$. Note that $\|\mathcal{W}'\| < \|\mathcal{W}\|$. Then $\mathcal{W}$ is not valid $\iff$ $\mathcal{W}'$ is not valid $\iff$ (by the inductive hypothesis) the main configuration $(\ell_0, \mathit{Lab}, \mathcal{W}', \texttt{false})$ leads to acceptance $\iff$ (by construction of `checkFalse`) the main configuration $(\ell_0, \mathit{Lab}, \mathcal{W}, \texttt{false})$ leads to acceptance.
- $\psi = \langle X \rangle \psi'$ with $X \in \{\overline{B}, \overline{E}\}$. By Proposition 25(1), $\mathcal{K}, \rho \models \langle X \rangle \psi'$ if and only if there exists an X-witness ρ' of ρ for $(\mathcal{K}, \varphi)$ such that $\mathcal{K}, \rho' \models \psi'$. Then (2.) of Claim 49 directly follows from the following chain of equivalences: $\mathcal{W}$ is not valid $\iff$ either $\mathcal{W} \setminus \{(\psi, \rho)\}$ is not valid, or for each X-witness ρ' of ρ for $(\mathcal{K}, \varphi)$, $\{(\psi', \rho')\}$ is not valid $\iff$ for each X-witness ρ' of ρ for $(\mathcal{K}, \varphi)$, $(\mathcal{W} \setminus \{(\psi, \rho)\}) \cup \{(\psi', \rho')\}$ is not valid $\iff$ (by the inductive hypothesis) for each X-witness ρ' of ρ for $(\mathcal{K}, \varphi)$, the main configuration $(\ell_0, \mathit{Lab}, (\mathcal{W} \setminus \{(\psi, \rho)\}) \cup \{(\psi', \rho')\}, \texttt{false})$ leads to acceptance $\iff$ (by construction of the procedure `checkFalse`) the main configuration $(\ell_0, \mathit{Lab}, \mathcal{W}, \texttt{false})$ leads to acceptance.

This concludes the proof of Claim 49. □

By exploiting Claim 49, we now prove the following result, which concludes the proof of Proposition 26.

Claim 50. The ATM `check` accepts an input $(\mathcal{K}, \varphi)$ if and only if $\mathcal{K} \models \varphi$.

Proof. Let us fix an input $(\mathcal{K}, \varphi)$ and an $\mathrm{A\overline{A}}$-labeling Lab for $(\mathcal{K}, \varphi)$. A *Lab-guessing* for $(\mathcal{K}, \varphi)$ is a well-formed set $\mathcal{W}$ for $(\mathcal{K}, \varphi)$ which minimally satisfies the following conditions for all states s of $\mathcal{K}$:

- for all certificates ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s_0$, $(\varphi, \rho) \in \mathcal{W}$;
- for all $\langle A \rangle \psi \in \mathit{Lab}(s)$ (respectively, $\langle \overline{A} \rangle \psi \in \mathit{Lab}(s)$), there is a certificate ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s$ (respectively, $\mathrm{lst}(\rho) = s$) such that $(\psi, \rho) \in \mathcal{W}$;
- for all $[A]\psi \in \mathit{Lab}(s)$ (respectively, $[\overline{A}]\psi \in \mathit{Lab}(s)$) and for all certificates ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s$ (respectively, $\mathrm{lst}(\rho) = s$), $(\psi, \rho) \in \mathcal{W}$.

Evidently, by construction of the procedure `check`, for each input $(\mathcal{K}, \varphi)$, it holds that:

- (*) `check` accepts $(\mathcal{K}, \varphi)$ $\iff$ there exists an $\mathrm{A\overline{A}}$-labeling Lab and a Lab-guessing $\mathcal{W}$ for $(\mathcal{K}, \varphi)$ such that, for all $(\psi, \rho) \in \mathcal{W}$, the main configuration $(\ell_0, \mathit{Lab}, \{(\psi, \rho)\}, \texttt{true})$ leads to acceptance.

Let us fix an input $(\mathcal{K}, \varphi)$. First we assume that $\mathcal{K} \models \varphi$. Let Lab be the *valid* $\mathrm{A\overline{A}}$-labeling defined as follows for all states s: for all $\psi \in \mathrm{A\overline{A}}(\varphi)$, $\psi \in \mathit{Lab}(s)$ if and only if $\mathcal{K}, s \models \psi$. By Theorem 23, there exists a Lab-guessing $\mathcal{W}$ for $(\mathcal{K}, \varphi)$ such that for all $(\psi, \rho) \in \mathcal{W}$, $\mathcal{K}, \rho \models \psi$. By Claim 49, for all $(\psi, \rho) \in \mathcal{W}$, the main configuration $(\ell_0, \mathit{Lab}, \{(\psi, \rho)\}, \texttt{true})$ leads to acceptance. Hence, by (*), `check` accepts $(\mathcal{K}, \varphi)$.

For the converse direction, let us assume that `check` accepts $(\mathcal{K}, \varphi)$. By (*), there exists an $\mathrm{A\overline{A}}$-labeling Lab and a Lab-guessing $\mathcal{W}$ for $(\mathcal{K}, \varphi)$ such that, for all $(\psi, \rho) \in \mathcal{W}$, the main configuration $(\ell_0, \mathit{Lab}, \{(\psi, \rho)\}, \texttt{true})$ leads to acceptance. First we show that Lab is valid.

We fix a state s and a formula $\psi \in \mathit{Lab}(s)$. We need to prove that $\mathcal{K}, s \models \psi$. The proof is by induction on the nesting depth $d_{\mathrm{A\overline{A}}}(\psi)$ of modalities $\langle \mathrm{A} \rangle$, $\langle \overline{\mathrm{A}} \rangle$, $[A]$, and $[\overline{A}]$ in ψ. Assume that $\psi = [A]\psi'$ for some ψ' (the other cases, where either $\psi = \langle \mathrm{A} \rangle \psi'$, or $\psi = \langle \overline{\mathrm{A}} \rangle \psi'$ or $\psi = [\overline{A}]\psi'$ are similar). By definition of Lab-guessing, for each certificate ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s$, $(\psi', \rho) \in \mathcal{W}$. Moreover, by the inductive hypothesis, one can assume that $\mathit{Lab}_{\{(\psi', \rho)\}}$ is valid (note that for the base case, i.e., when ψ' does not contain occurrences of modalities $\langle \mathrm{A} \rangle$, $\langle \overline{\mathrm{A}} \rangle$, $[A]$, and $[\overline{A}]$, $\mathit{Lab}_{\{(\psi', \rho)\}}$ is trivially valid). By hypothesis, the main configuration $(\ell_0, \mathit{Lab}, \{(\psi', \rho)\}, \texttt{true})$ leads to acceptance. By Claim 49, for each certificate ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s$, it holds $\mathcal{K}, \rho \models \psi'$. Thus, by Theorem 23, we obtain that $\mathcal{K}, s \models \psi$. Hence Lab is valid.

Now, by definition of Lab-guessing, for each certificate ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s_0$, $(\varphi, \rho) \in \mathcal{W}$. Thus, by hypothesis, by Claim 49, and by Theorem 23, we have $\mathcal{K} \models \varphi$. This concludes the proof of the claim. □

Proposition 26 has been proved. □

A.4. Proof of Theorem 45

Proof. Given a regular expression r with $\mathcal{L}(r) \subseteq \Sigma^*$, let us define the finite Kripke structure $\mathcal{K} = (\Sigma, \{s_0\} \cup \Sigma, R, \mu, s_0)$, where $s_0 \notin \Sigma$, $\mu(s_0) = \emptyset$, for $c \in \Sigma$, $\mu(c) = \{c\}$, and $R = \{(s_0, c) \mid c \in \Sigma\} \cup \{(c, c') \mid c, c' \in \Sigma\}$. It is easy to see that $\mathcal{L}(r) = \Sigma^* \iff \mathcal{K} \models \top \cdot \overline{r}$, where $\overline{r}$ is a RE over Σ, *syntactically* equal to r. Note that even though if r and $\overline{r}$ are syntactically equal, r is a regular expression defining a finitary language over Σ, whereas $\overline{r}$ defines a finitary language *over* 2^Σ (see Section 2). The different notations r and $\overline{r}$ are kept to avoid confusion between the two different semantics.

We show by induction on the structure of r that, for all $w \in \Sigma^*$, $w \in \mathcal{L}(r) \iff \mathcal{K}, w \models \overline{r}$. The thesis follows as $\mathcal{K}, w \models \overline{r}$ if and only if $\mathcal{K}, s_0 \cdot w \models \top \cdot \overline{r}$.

Case $r = \varepsilon$. We have $w \in \mathcal{L}(\varepsilon)$ if and only if $w = \varepsilon$, if and only if $\mu(w) \in \mathcal{L}(\overline{\varepsilon}) = \{\varepsilon\}$, if and only if $\mathcal{K}, w \models \overline{\varepsilon}$.

Case $r = c \in \Sigma$. We have $w \in \mathcal{L}(c)$ if and only if $w = c$, thus $\mu(w) = \{c\} \in \mathcal{L}(\overline{c})$, and $\mathcal{K}, w \models \overline{c}$. Conversely, if $\mathcal{K}, w \models \overline{c}$ we have $\mu(w) \in \mathcal{L}(\overline{c}) = \{A \in 2^\Sigma \mid c \in A\}$. In particular $|w| = 1$. Moreover, by definition of μ, $\mu(w)$ is a singleton, hence $\mu(w) = \{c\}$. By definition of $\mathcal{K}$ we get $w = c$, thus $w \in \mathcal{L}(c)$.

Case $r = r_1 \cdot r_2$. We have $w \in \mathcal{L}(r_1 \cdot r_2)$ if and only if $w = w_1 \cdot w_2$, with $w_1 \in \mathcal{L}(r_1)$ and $w_2 \in \mathcal{L}(r_2)$. By applying the inductive hypothesis, $\mathcal{K}, w_1 \models \overline{r_1}$ and $\mathcal{K}, w_2 \models \overline{r_2}$, thus $\mu(w_1) \in \mathcal{L}(\overline{r_1})$ and $\mu(w_2) \in \mathcal{L}(\overline{r_2})$. It follows that $\mu(w) = \mu(w_1) \cdot \mu(w_2) \in \mathcal{L}(\overline{r_1}) \cdot \mathcal{L}(\overline{r_2}) = \mathcal{L}(\overline{r_1 \cdot r_2})$, namely $\mathcal{K}, w \models \overline{r_1 \cdot r_2}$. Conversely, $\mu(w) \in \mathcal{L}(\overline{r_1 \cdot r_2}) = \mathcal{L}(\overline{r_1}) \cdot \mathcal{L}(\overline{r_2})$. Hence $\mu(w_1) \in \mathcal{L}(\overline{r_1})$ and $\mu(w_2) \in \mathcal{L}(\overline{r_2})$, for some $w_1 \cdot w_2 = w$. By the inductive hypothesis, $w_1 \in \mathcal{L}(r_1)$ and $w_2 \in \mathcal{L}(r_2)$, hence $w \in \mathcal{L}(r_1 \cdot r_2)$.

Case $r = r_1 \cup r_2$. We have $w \in \mathcal{L}(r_1 \cup r_2)$ if and only if $w \in \mathcal{L}(r_i)$ for some $i = 1, 2$. By the inductive hypothesis this is true if and only if $\mathcal{K}, w \models \overline{r_i}$, if and only if $\mu(w) \in \mathcal{L}(\overline{r_i})$, if and only if $\mu(w) \in \mathcal{L}(\overline{r_1 \cup r_2})$, if and only if $\mathcal{K}, w \models \overline{r_1 \cup r_2}$.

Case $r = r_1^*$. The thesis trivially holds if $w = \varepsilon$. Let us now assume $w \neq \varepsilon$. We have $w \in \mathcal{L}(r_1^*)$ if and only if, for some $t \geq 1$, $w = w_1 \cdots w_t$ and $w_\ell \in \mathcal{L}(r_1)$ for all $1 \leq \ell \leq t$. By the inductive hypothesis, $\mathcal{K}, w_\ell \models \overline{r_1}$, thus $\mu(w_\ell) \in \mathcal{L}(\overline{r_1})$, and $\mu(w) \in \mathcal{L}(\overline{r_1^*})$. We conclude that $\mathcal{K}, w \models \overline{r_1^*}$. Conversely, $\mu(w) \in \mathcal{L}(\overline{r_1^*}) = (\mathcal{L}(\overline{r_1}))^*$, hence it must be the case that, for some $t \geq 1$, $w = w_1 \cdots w_t$ and $\mu(w_\ell) \in \mathcal{L}(\overline{r_1})$ for all $1 \leq \ell \leq t$. By the inductive hypothesis, $w_\ell \in \mathcal{L}(r_1)$, hence $w \in \mathcal{L}(r_1^*)$. Finally, by observing that $\mathcal{K}$ can be built by using logarithmic working space, the thesis follows. □

References

[1] J.F. Allen, Maintaining knowledge about temporal intervals, Commun. ACM 26 (11) (1983) 832–843.

[2] C. Baier, J. Katoen, Principles of Model Checking, MIT Press, 2008.

[3] L. Bozzelli, A. Molinari, A. Montanari, A. Peron, An in-depth investigation of interval temporal logic model checking with regular expressions, in: SEFM, in: LNCS, vol. 10469, Springer, 2017, pp. 104–119.

[4] L. Bozzelli, A. Molinari, A. Montanari, A. Peron, On the complexity of model checking for syntactically maximal fragments of the interval temporal logic HS with regular expressions, in: GandALF, in: EPTCS, vol. 256, 2017, pp. 31–45.

[5] L. Bozzelli, A. Molinari, A. Montanari, A. Peron, On the complexity of model checking for syntactically maximal fragments of the interval temporal logic HS with regular expressions, University of Udine, Italy, 2017, Tech. rep., https://www.dimi.uniud.it/la-ricerca/pubblicazioni/preprints/3.2017/.

[6] L. Bozzelli, A. Molinari, A. Montanari, A. Peron, P. Sala, Interval temporal logic model checking: the border between good and bad HS fragments, in: IJCAR, in: LNAI, vol. 9706, Springer, 2016, pp. 389–405.

[7] L. Bozzelli, A. Molinari, A. Montanari, A. Peron, P. Sala, Interval vs. point temporal logic model checking: an expressiveness comparison, in: FSTTCS 2016, in: LIPIcs. Schloss Dagstuhl—Leibniz-Zentrum fuer Informatik, vol. 65, 2016, pp. 26:1–14.

[8] L. Bozzelli, A. Molinari, A. Montanari, A. Peron, P. Sala, Model checking the logic of Allen's relations meets and started-by is $\mathbf{P}^{\mathbf{NP}}$-complete, in: GandALF, in: EPTCS, vol. 226, 2016, pp. 76–90.

[9] L. Bozzelli, H. van Ditmarsch, S. Pinchinat, The complexity of one-agent refinement modal logic, Theor. Comput. Sci. 603 (C) (2015) 58–83.

[10] D. Bresolin, D. Della Monica, V. Goranko, A. Montanari, G. Sciavicco, The dark side of interval temporal logic: marking the undecidability border, Ann. Math. Artif. Intell. 71 (1–3) (2014) 41–83.

[11] D. Bresolin, V. Goranko, A. Montanari, P. Sala, Tableau-based decision procedures for the logics of subinterval structures over dense orderings, J. Log. Comput. 20 (1) (2010) 133–166.

[12] D. Bresolin, V. Goranko, A. Montanari, G. Sciavicco, Propositional interval neighborhood logics: expressiveness, decidability, and undecidable extensions, Ann. Pure Appl. Log. 161 (3) (2009) 289–304.

[13] A.K. Chandra, D.C. Kozen, L.J. Stockmeyer, Alternation, J. ACM 28 (1) (1981) 114–133.

[14] E.A. Emerson, J.Y. Halpern, "Sometimes" and "not never" revisited: on branching versus linear time temporal logic, J. ACM 33 (1) (1986) 151–178.

[15] J. Esparza, D. Hansel, P. Rossmanith, S. Schwoon, Efficient algorithms for model checking pushdown systems, in: CAV, in: LNCS, vol. 1855, Springer, 2000, pp. 232–247.

[16] J. Ferrante, C. Rackoff, A decision procedure for the first order theory of real addition with order, SIAM J. Comput. 4 (1) (1975) 69–76.

[17] M. Garey, D. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, 1979.

[18] F. Giunchiglia, P. Traverso, Planning as model checking, in: ECP, in: LNCS, vol. 1809, Springer, 1999, pp. 1–20.

[19] M. Gligoric, R. Majumdar, Model checking database applications, in: TACAS, in: LNCS, vol. 7795, Springer, 2013, pp. 549–564.

[20] J.Y. Halpern, Y. Shoham, A propositional modal logic of time intervals, J. ACM 38 (4) (1991) 935–962.

[21] O. Kupferman, N. Piterman, M.Y. Vardi, From liveness to promptness, Form. Methods Syst. Des. 34 (2) (2009) 83–103.

[22] M. Leucker, C. Sánchez, Regular linear temporal logic, in: ICTAC, in: LNCS, vol. 4711, Springer, 2007, pp. 291–305.

[23] A. Lomuscio, J. Michaliszyn, An epistemic Halpern-Shoham logic, in: IJCAI, in: IJCAI/AAAI, 2013, pp. 1010–1016.

[24] A. Lomuscio, J. Michaliszyn, Decidability of model checking multi-agent systems against a class of EHS specifications, in: Frontiers in Artificial Intelligence and Applications, in: ECAI, vol. 263, IOS Press, 2014, pp. 543–548.

[25] A. Lomuscio, J. Michaliszyn, Model checking multi-agent systems against epistemic HS specifications with regular expressions, in: KR, AAAI Press, 2016, pp. 298–308.

[26] A. Lomuscio, F. Raimondi, MCMAS: a model checker for multi-agent systems, in: TACAS, in: LNCS, vol. 3920, Springer, 2006, pp. 450–454.

[27] J. Marcinkowski, J. Michaliszyn, The undecidability of the logic of subintervals, Fundam. Inform. 131 (2) (2014) 217–240.

[28] R. Mateescu, P.T. Monteiro, E. Dumas, H. de Jong, CTRL: extension of CTL with regular expressions and fairness operators to verify genetic regulatory networks, Theor. Comput. Sci. 412 (26) (2011) 2854–2883.

[29] A. Molinari, A. Montanari, A. Murano, G. Perelli, A. Peron, Checking interval properties of computations, Acta Inform. 53 (6–8) (2016) 587–619.

[30] A. Molinari, A. Montanari, A. Peron, Complexity of ITL model checking: some well-behaved fragments of the interval logic HS, in: TIME, IEEE, 2015, pp. 90–100.

[31] A. Molinari, A. Montanari, A. Peron, A model checking procedure for interval temporal logics based on track representatives, in: LIPIcs. Schloss Dagstuhl—Leibniz-Zentrum fuer Informatik, in: CSL, vol. 41, 2015, pp. 193–210.

[32] A. Molinari, A. Montanari, A. Peron, P. Sala, Model checking well-behaved fragments of HS: the (almost) final picture, in: KR, AAAI Press, 2016, pp. 473–483.

[33] A. Montanari, Interval temporal logics model checking, in: TIME, IEEE, 2016, p. 2.

[34] B. Moszkowski, Reasoning about digital circuits, Ph.D. thesis, Stanford University, Stanford, CA, 1983.

[35] A. Pnueli, The temporal logic of programs, in: FOCS, IEEE, 1977, pp. 46–57.

[36] P. Roeper, Intervals and tenses, J. Philos. Log. 9 (1980) 451–469.

[37] P. Schnoebelen, Oracle circuits for branching-time model checking, in: ICALP, in: LNCS, vol. 2719, Springer, 2003, pp. 790–801.

[38] Y. Venema, Expressiveness and completeness of an interval tense logic, Notre Dame J. Form. Log. 31 (4) (1990) 529–547.